

Probabilistic-Based Rate Adaptation for IEEE 802.11 WLANs

Xi Chen, Daji Qiao

Department of Electrical and Computer Engineering
Iowa State University, Ames, IA 50011
{leon6827, daji}@iastate.edu

Jeonggyun Yu, Sunghyun Choi

School of Electrical Engineering and INMC
Seoul National University, Seoul, 151-744, Korea
jgyu@mwnl.snu.ac.kr, schoi@snu.ac.kr

Abstract—Collision awareness has been recognized as a critical component for effective rate adaptation schemes. Recently, several collision-aware rate adaptation schemes have been proposed for IEEE 802.11 Wireless LANs (WLANs), such as CARA (Collision Aware Rate Adaptation) and RRAA (Robust Rate Adaptation Algorithm). These schemes are able to distinguish between channel-error-induced and collision-induced frame losses via adaptive and appropriate usage of RTS/CTS; hence the multiple transmission rates provided by 802.11 physical layers (PHYs) may be fully exploited. In this paper, we propose a unique collision-aware rate adaptation scheme, called PBRA (Probabilistic-Based Rate Adaptation). The key ideas of PBRA include (i) probabilistic-based adaptive usage of RTS/CTS, which is in direct contrast to trial-based RTS Probing in CARA and window-based adaptive usage of RTS/CTS in RRAA; and (ii) threshold-based rate adjustment, which allows a station to make more appropriate rate adjustment decisions, thanks to its accurate estimation of the channel-error-induced frame loss ratio. Simulation results show that PBRA clearly outperforms all other testing schemes (including CARA and RRAA), particularly in random topology networks with fading wireless channels.

I. INTRODUCTION

IEEE 802.11 is the most popular Wireless LAN (WLAN) system in the world today, and it is likely to play an important role in the next generation of wireless and mobile communication systems. IEEE 802.11 physical layers (PHYs) provide multiple transmission rates by employing different modulation and coding schemes. For example, the 802.11g PHY supports 12 rates from 1 to 54 Mbps. To exploit such multi-rate capability, the transmitting station should adjust its transmission rate dynamically to the time-varying and location-dependent channel conditions, either with or without explicit feedback from the receiver [1]–[7]. Since the former category of rate adaptation schemes (referred to as *close-loop* schemes) do not conform to the 802.11 standard, we focus on *open-loop* schemes with which the transmitter makes the rate adjustment decisions without any feedback from the receiver.

Conventional open-loop rate adaptation schemes, such as the ARF [1], monitor the frame loss events and adjust the transmission rate accordingly. Ideally, rate adjustment shall be triggered by *channel-error-induced* frame losses only. However, these schemes are not able to differentiate them from *collision-induced* frame losses, and hence may malfunction severely when there are many frame collisions in the network. CARA [4] is one of the first schemes proposed to handle the problem discussed above. It uses a trial-based RTS Probing mechanism to identify causes of frame

losses. The key idea of RTS Probing is to enable the RTS/CTS exchange to probe the wireless channel condition whenever a data transmission failure occurs. CARA's rate adjustment component is based on the ARF, which is aggressive in decreasing the transmission rate (i.e., upon observation of two successive failures), and is relatively conservative in increasing the rate (i.e., upon observation of 10 consecutive successes). CARA-RI [5] is an enhancement to CARA. It only resets the consecutive success count when the transmission rate starts decreasing. Thus, CARA-RI becomes more aggressive in increasing the transmission rate and yields higher throughput than the original CARA (referred to as CARA-BASIC in this paper).

RRAA [3] is another recently proposed scheme which employs a window-based approach to use RTS/CTS adaptively. It has the following two limitations. Firstly, its window-based adaptive usage of RTS/CTS may be less responsive to variation of the network condition. For example, if the window size is too large while the number of contending stations has been reduced, it may incur many unnecessary RTS transmissions. Secondly, when calculating the frame loss ratio, RRAA does not distinguish between channel-error-induced and collision-induced losses, which may lead to incorrect rate adjustment decisions and consequently unsatisfactory performance.

In this paper, we propose a probabilistic-based rate adaptation scheme, called PBRA. The key idea of PBRA is twofold. Firstly, PBRA decides whether to use RTS/CTS for each data transmission attempt based on a probability P_{rts}^* , which minimizes the expected time to deliver the current data frame successfully. In order to determine P_{rts}^* , every station estimates the current collision probability (P_c) upon observation of the outcome of each transmission attempt, and then uses a mathematical model to derive the corresponding P_{rts}^* . This is in direct contrast to the trial-based RTS Probing in CARA and the window-based adaptive usage of RTS/CTS in RRAA. Secondly, PBRA differentiates channel-error-induced frame losses from collision-induced losses, thus being able to make more appropriate rate adjustment decisions.

The rest of the paper is organized as follows. Relevant issues and details of PBRA are described in Section II. Section III presents the simulation results, and finally, the paper concludes in Section IV.

II. PBRA: PROBABILISTIC-BASED RATE ADAPTATION

Our proposed PBRA scheme consists of the following two parts: probabilistic-based adaptive usage of RTS/CTS and threshold-based rate adjustment.

The research reported in this paper was supported in part by the Information Infrastructure Institute (iCube) of Iowa State University and the National Science Foundation under Grants No. CNS 0520102 and CNS 0540293.

A. Probabilistic-Based Adaptive Usage of RTS/CTS

Adaptive usage of the RTS/CTS has been recognized as an effective way to deal with hidden stations in 802.11 networks and to improve the throughput performance. CARA-like schemes use the RTS/CTS exchange adaptively via RTS Probing. The key idea of RTS Probing is to enable the RTS/CTS exchange (to probe the wireless channel condition) whenever a DATA transmission failure occurs. With RTS Probing, CARA demonstrates significant throughput improvement over ARF. However, if there are no hidden stations in the network, it may not be a good idea to always probe the channel condition upon a transmission failure. Moreover, CARA does not use RTS/CTS exchange for the first attempt of a DATA frame transmission, which is not reasonable if there are hidden stations in the network. As a result, such a trial-based RTS Probing mechanism may not perform well under certain circumstances.

In comparison, PBRA employs a probabilistic-based adaptive usage of RTS/CTS exchange. With PBRA, each station maintains a parameter, P_c ($0 \leq P_c \leq 1$), as an estimate of the current collision probability in the network regardless of the colliding frames being RTS or DATA. When P_c is large, it means that the contention is severe or hidden nodes exist in the network. On the other hand, when P_c is small, the chance that a frame transmission experiences collision is low. The P_c value is updated upon observation of the outcome of each transmission attempt. Table I lists two ways of attempting a DATA frame transmission (i.e., with or without RTS/CTS), possible outcomes of each transmission attempt, and the corresponding actions on updating the P_c value. Here, α ($\alpha \leq \alpha_{\max}$) and β ($\beta \geq \beta_{\min}$) are the design parameters used to update P_c at runtime. Then, based on the P_c value, PBRA calculates P_{rts}^* – the probability it uses to determine whether to use RTS/CTS for the next transmission attempt.

TABLE I

DATA TRANSMISSION ATTEMPTS, OUTCOMES, AND CORRESPONDING ACTIONS ON UPDATING P_c

DATA Frame Transmission Attempt	Outcome of Transmission Attempt	Action on Updating P_c
DATA with RTS	RTS Fail	$P_c = \min(1, P_c \cdot \alpha_{\max})$
	RTS Succ, DATA Fail RTS Succ, DATA Succ	$P_c = P_c / \beta_{\min}$
DATA without RTS	DATA Fail	$P_c = P_c \cdot \alpha$
	DATA Succ	$P_c = P_c / \beta$

In the following, we first discuss in detail how to update the P_c value at runtime, and then describe briefly the model we use to calculate P_{rts}^* .

1) *How to update P_c :* In general, upon observation of a transmission failure, P_c should be increased, while a successful transmission indicates no collision to the current transmission attempt, and hence P_c should be decreased. More specifically, there are four different cases as shown below.

• Case I: RTS Failure

We assume that the frame loss probability due to channel error for small-size frames is negligible. So an RTS failure indicates that the collision problem is severe under the current situation. We hence multiply P_c by α_{\max} .

• Case II: DATA Failure (without RTS/CTS support)

Without RTS/CTS support, when a DATA transmission fails, we are not sure about the cause of failure. In this case, P_c is multiplied by α and the value of α depends on how confident we are about collision is the cause of failure. Specifically, when the DATA transmission time is the same as the RTS transmission time, P_c is multiplied by α_{\max} (i.e., $\alpha = \alpha_{\max}$). In general, as the DATA transmission time increases, the probability that DATA failure is caused by channel error increases, which means that we are less confident about collision is the cause of failure, and consequently, a smaller α should be used. The correlation between α and α_{\max} values will be discussed at the end of this section.

• Case III: RTS Success

In this case, P_c is simply decreased by β_{\min} regardless whether the subsequent DATA transmission succeeds or not. This is because an RTS/CTS exchange has already reserved the wireless channel and guarantees no collision to the subsequent DATA transmission. Hence, the result of DATA transmission does not affect our estimation of the current collision probability in the network.

• Case IV: DATA Success (without RTS/CTS support)

Without RTS/CTS support, when a DATA transmission succeeds, P_c is decreased by β which is a monotonously-increasing function of the DATA transmission time. β_{\min} is used when the DATA transmission time is the same as the RTS transmission time. This is because a successful DATA frame transmission with longer transmission time and no preceding RTS usually indicates smaller chance of collision, and hence P_c should be decreased more sharply. The correlation between β and β_{\min} values will be discussed at the end of this section.

2) *How to calculate P_{rts}^* :* We focus on a given station and derive the relation between P_c and P_{rts} which is the probability of using RTS/CTS for a DATA transmission attempt. Table II lists the time duration d and probability P for each of the five DATA transmission attempt outcomes. Expressions for time durations are omitted due to space limitation. P_e^{data} is the DATA error probability (due to deteriorated channel condition) and P_c^{data} and P_c^{rts} are the collision probability for DATA frames (when RTS/CTS is not used) and RTS frames respectively.

The expected time to deliver a DATA frame successfully, denoted as $E[T]$, is a function of P_{rts} , P_c^{rts} , and P_c^{data} , and can be obtained from the following recursive equation:

$$E[T] = P_1(d_1 + E[T]) + P_2(d_2 + E[T]) + P_3d_3 + P_4(d_4 + E[T]) + P_5d_5. \quad (1)$$

The goal is to find the optimal P_{rts} value that minimizes $E[T]$. Since we have

$$\begin{cases} 1 - P_c^{rts} = (1 - \tau) \frac{T_{rts} + \overline{T}_{data}}{aSlotTime} \cdot N_{hidden} + N_{nbr}, \\ 1 - P_c^{data} = (1 - \tau) \frac{2\overline{T}_{data}}{aSlotTime} \cdot N_{hidden} + N_{nbr}, \end{cases} \quad (2)$$

hence

$$\frac{\ln(1 - P_c^{rts})}{\ln(1 - P_c^{data})} \approx \frac{T_{rts} + \overline{T}_{data}}{2\overline{T}_{data}}, \quad (3)$$

when there exist hidden nodes in the network (i.e., $N_{hidden} \geq 1$).

TABLE II
DATA TRANSMISSION ATTEMPTS, OUTCOMES, AND THE CORRESPONDING TIME DURATIONS AND PROBABILITIES

DATA Frame Transmission Attempt	Outcome of Transmission Attempt	Time Duration	Probability
DATA with RTS	RTS Fail	d_1	$P_1 = P_{rts}P_c^{rts}$
	RTS Succ, DATA Fail	d_2	$P_2 = P_{rts}(1 - P_c^{rts})P_e^{data}$
	RTS Succ, DATA Succ	d_3	$P_3 = P_{rts}(1 - P_c^{rts})(1 - P_e^{data})$
DATA without RTS	DATA Fail	d_4	$P_4 = (1 - P_{rts})[1 - (1 - P_c^{data})(1 - P_e^{data})]$
	DATA Succ	d_5	$P_5 = (1 - P_{rts})(1 - P_c^{data})(1 - P_e^{data})$

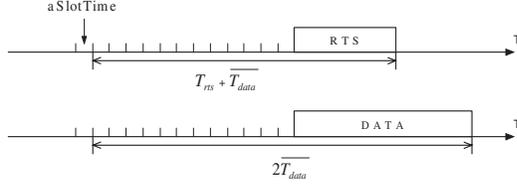


Fig. 1. Illustration of vulnerable periods for RTS transmission and DATA transmission when there are hidden stations in the network.

Moreover, we know that

$$P_c = P_{rts}P_c^{rts} + (1 - P_{rts})P_c^{data}. \quad (4)$$

Combining (1), (3) and (4), $E[T]$ becomes a function of P_{rts} and P_c . Here, τ is the probability that a station transmits in a randomly chosen time slot, \overline{T}_{data} is the average DATA transmission time, N_{hidden} and N_{nbr} are the number of hidden stations and neighbors of the station of our interest, respectively. (2) is obtained based on the observation that, in order to transmit an RTS or DATA frame successfully, we require no transmission attempt from other hidden stations during the respective *vulnerable period*, as illustrated in Figure 1. Consequently, we have:

$$P_{rts}^* = \arg \min_{P_{rts}} E[T]. \quad (5)$$

Numerical results for P_{rts}^* are plotted in Figure 2. It is interesting to see that, given P_c , DATA frame size and DATA transmission rate, P_{rts}^* is actually a boolean function (i.e., $P_{rts}^* = 0$ or 1). For example, as shown in the figure, if a station wants to transmit a 1500-byte DATA frame at 2 Mbps, and if the current P_c is larger than 0.1, P_{rts}^* is equal to 1, meaning that RTS/CTS exchange should always be used in this situation.

Before communication starts, each station pre-computes a P_{rts}^* table that is indexed by P_c and the DATA transmission time. At runtime, for each DATA transmission attempt, a station simply looks up this table for the P_{rts}^* value based on the most updated P_c and the calculated DATA transmission time, then decides whether to use RTS/CTS to aid this DATA transmission attempt.

The relationship between β (α) and β_{min} (α_{max}) can be derived as follows. Let's take β and β_{min} for example. Assume that the current estimation of collision probability is P_c^{curr} . If the next DATA transmission (without RTS/CTS support) fails, the estimation of collision probability should be decreased by β . Based on the second equation of (2), we have

$$\frac{\ln(1 - P_c^{curr}/\beta_{min})}{\ln(1 - P_c^{curr}/\beta)} = \frac{T_{rts}}{T_{data}}, \quad (6)$$

which indicates how to determine β according to the failed DATA transmission time and β_{min} . α can also be calculated in a similar way.

B. Threshold-Based Rate Adjustment

Ideally, rate adjustment shall only be triggered by channel-error-induced DATA transmission failures. PBRA's rate adjustment mechanism is motivated by that employed in RRAA, which

is based on the frame loss count during an estimation window. However, RRAA interprets all frame losses as channel-error-induced losses, which is not true even with the help of adaptive usage of RTS/CTS.

To deal with this problem, we identify *ambiguous frame losses* whose cause is not clear, and then calculate the lower and upper bounds for the channel-error-induced frame loss ratio. As shown in Table III, when a DATA transmission fails without preceding RTS, we are not sure about the cause of frame loss and such a frame loss is deemed *ambiguous* in PBRA.

TABLE III
OUTCOMES OF DATA TRANSMISSION ATTEMPTS AND THEIR IMPLICATIONS ON CHANNEL CONDITION

DATA Frame Transmission Attempt	Outcome of Transmission Attempt	Channel-Error Induced DATA Loss?
DATA with RTS	RTS Fail	N/A
	RTS Succ, DATA Fail	Yes
	RTS Succ, DATA Succ	No
DATA without RTS	DATA Fail	Ambiguous
	DATA Succ	No

We use $loss_{rtson}$ and $succ_{rtson}$ to denote the frame loss and success counts during the estimation window when RTS/CTS is used for DATA transmission attempts, and use $loss_{rtsoff}$ and $succ_{rtsoff}$ to denote the frame loss and success counts when RTS/CTS is not used. The lower and upper bounds for the channel-error-induced frame loss ratio are, respectively,

$$\mathcal{L}_{lower} = \frac{loss_{rtson}}{loss_{rtson} + loss_{rtsoff} + succ_{rtson} + succ_{rtsoff}} \quad (7)$$

and

$$\mathcal{L}_{upper} = \frac{loss_{rtson} + loss_{rtsoff}}{loss_{rtson} + loss_{rtsoff} + succ_{rtson} + succ_{rtsoff}}, \quad (8)$$

and they are updated every time an estimation window elapses.

Let $\mathcal{L}^*(r)$ denote the *critical frame loss ratio* for the transmission rate r , which is defined as the frame loss ratio with which the throughput at the current transmission rate r is equal to the loss-free throughput achieved at the transmission rate right lower than r . Figure 3 reveals the relation between frame loss ratio and throughput. Dashed lines in the figure illustrate how to obtain the critical frame loss ratio for $r = 5.5$ Mbps, which is about $\mathcal{L}^*(5.5) = 0.479$. We observe that the throughput for each transmission rate decreases almost linearly as the frame loss ratio increases, which provides convenience in our calculation of the $\mathcal{L}^*(r)$ value.

PBRA's threshold-based rate adjustment scheme is illustrated in Figure 4: assume that the current transmission rate is r , then the rate is decreased when $\mathcal{L}_{lower} \geq \mathcal{L}^*(r)$; the rate is increased when $\mathcal{L}_{upper} \leq \mathcal{L}^*(r)/\gamma$ where $\gamma > 1$ is a tunable parameter; otherwise, the rate remains unchanged.

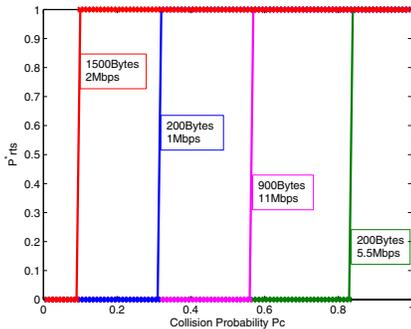


Fig. 2. P_{rts}^* vs. P_c

III. PERFORMANCE EVALUATION

In this section, we use the ns-2 simulator [12] to evaluate the performance of our proposed PBRA scheme. We simulate an infrastructure-based IEEE 802.11b system with various network topologies. 20 dBm transmission power is used for all stations and the background noise level is set to -96 dBm. We use the empirical BER vs. SNR curves provided by Intersil [10], and a log-distance path-loss model with the path-loss exponent of 4 to simulate indoor wireless environments.

We evaluate the following rate adaptation schemes: ARF [1], CARA-BASIC [4], CARA-RI [5], RRAA [3], and PBRA. The schemes are compared with each other in terms of the aggregate system throughput (in Mbps). The parameters used in PBRA are: $\alpha_{max} = 3.0$, $\beta_{min} = 1.1$, $\gamma = 1.2$, and the estimation window is 0.5 seconds.

A. Star Topology

We first compare the schemes in star-topology networks with various number of contending stations, in order to study the collision effect on the system performance. In this setup, contending stations are evenly spaced on a circle centered at the AP with a radius of 40 meters, and an AWGN wireless channel model is assumed. There are no hidden nodes because the maximum transmission range at the lowest rate of 1 Mbps is 80 meters.

Simulation results are plotted in Figure 5(a). We first observe that the system throughput of ARF is severely degraded as the number of contending stations increases. This ill behavior can be explained by ARF's inability to differentiate collisions from channel errors. In consequence, a station may decrease its transmission rate over-aggressively and operate at lower transmission rates.

Secondly, when only a few stations are transmitting, ARF shows better performance than CARA-BASIC and CARA-RI due to the saving of RTS/CTS overheads. Thanks to its advanced threshold-based rate adjustment, PBRA yields higher throughput than both ARF and CARA-like schemes.

Thirdly, although the rate adjustment scheme in PBRA is similar to that in RRAA, PBRA performs better than RRAA for the following two reasons. First, RRAA interprets all frame losses as channel-error-induced losses, which is not true even with adaptive usage of RTS/CTS. In comparison, PBRA identifies *ambiguous frame losses* when estimating the channel-error-induced frame loss ratio, and compares its upper and lower bounds with critical loss ratios when making the rate adjustment decisions. Second, the probabilistic-based usage of RTS/CTS in PBRA is more

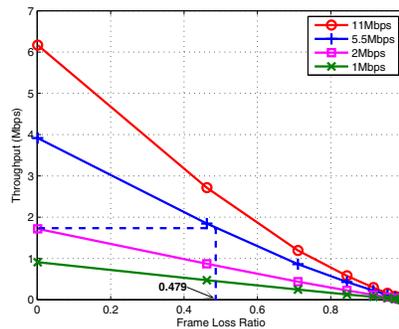


Fig. 3. Frame loss ratio vs. Throughput

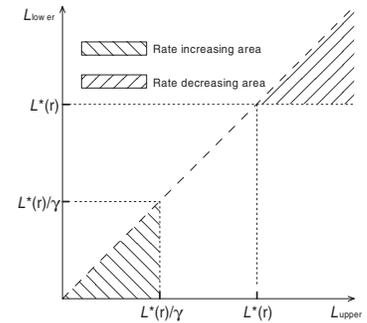


Fig. 4. PBRA's threshold-based rate adjustment

adaptive and responsive to variation of the network condition than the window-based usage of RTS/CTS in RRAA.

B. Hidden-Station Topology

We now evaluate the testing schemes in the presence of hidden stations. The simulated hidden station topology is set up as follows. Two groups of stations and the AP are placed along a straight line segment of 90 meters, which is larger than the maximum transmission range at the lowest rate of 1 Mbps (i.e., 80 meters). At each end of the segment, a group of stations are closely located together, and the AP is placed in the middle of the segment. Thus, AP can communicate with any station, while stations belonging to different groups are hidden from each other. An AWGN wireless channel model is assumed in this simulation.

We vary the group size from 1 to 10 (or equivalently, number of contending stations from 2 to 20). Each station randomly selects its data frame size (720 bytes on average), and transmits in a saturated manner to the AP. Thus frame collisions by hidden stations can be often observed at the AP. Simulation results are shown in Figure 5(b), where each point is averaged over 50 simulation runs.

We observe that PBRA is comparable with CARA-RI in hidden-station topology networks rather than performing better as in star-topology networks. This can be explained as follows. In our setup of hidden-station topology networks, the fraction of ambiguous frame losses is high. So the difference between \mathcal{L}_{lower} and \mathcal{L}_{upper} is large, which makes PBRA's rate adjustment scheme not very effective in this situation. Although PBRA shows its weaknesses in hidden-station topology networks, it still performs as well as CARA-RI, and outperforms all other testing schemes.

Another interesting observation is that RRAA yields even worse performance than ARF (with RTS/CTS support). Recall RRAA's design flaw discussed in Section I: when calculating the frame loss ratio, RRAA does not distinguish between channel-error-induced losses and collision-induced losses. Our setup of hidden-station topology networks, within which most of the frame losses are indeed caused by collisions, amplifies such design flaw. As a result, RRAA tends to make inappropriate rate adjustment decisions which lead to unsatisfactory performance.

C. Random Topology with Fading Channel

Now we evaluate and compare the performances of the testing schemes under more practical considerations in randomly-generated networks (i.e., random location and random DATA frame size for each station). In our setup, all stations are randomly

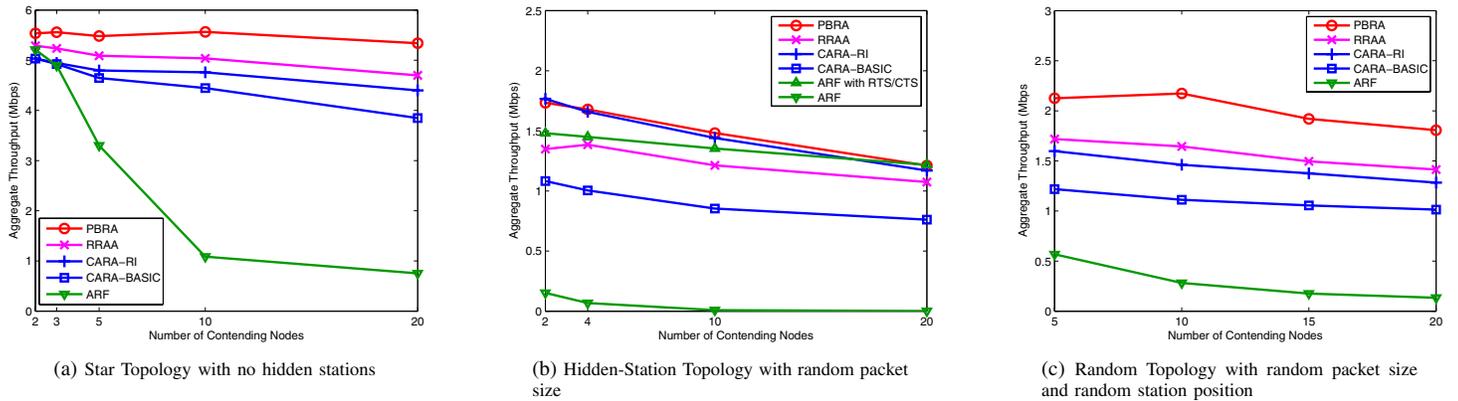


Fig. 5. Performance comparison of the testing schemes in various network topologies

placed within a circle around the AP with the radius of 80 meters. We assume a Ricean fading channel with Ricean K factor of 3 dB to describe the indoor fading channel environment. We vary the number of contending stations from 5 to 20. Simulation results are plotted in Figure 5(c), where each point is averaged over 50 runs.

As shown in the figure, PBRA clearly outperforms all other testing schemes, due to its probabilistic-based adaptive usage of RTS/CTS and proper estimation of the frame loss ratio when making the rate adjustment decisions. In particular, the probabilistic-based nature of PBRA's adaptive RTS/CTS usage seems to be a good fit for random-topology networks where hidden nodes may exist and channel conditions are less predictable due to fading effects. Simulation results in this subsection are particularly meaningful and encouraging because such network setup is practical and emulates the real network scenarios well.

D. Sensitivity Evaluation of α_{\max} and β_{\min}

The sensitivity of α_{\max} and β_{\min} values is evaluated in a randomly generated network which is similar to the network topology in Section III-C. In addition to 10 nodes randomly placed around the AP, we add two pairs of hidden nodes in order to guarantee the existence of hidden nodes. We fix one parameter (α_{\max} or β_{\min}) while varying the other to obtain the simulation results in Figure 6 where each point is averaged over 50 runs.

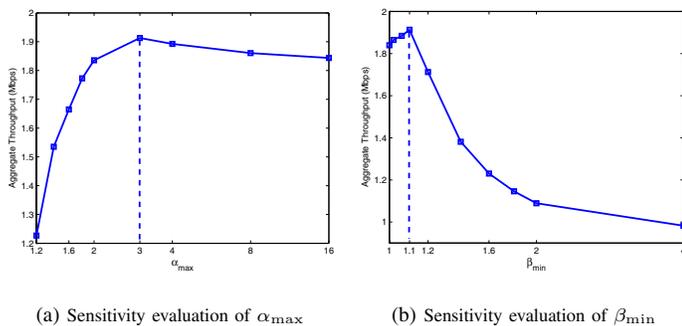


Fig. 6. Sensitivity evaluation of α_{\max} and β_{\min}

In Figure 6(a), β_{\min} is fixed to 1.1, and we vary the value of α_{\max} from 1.2 to 16.0. Heuristically, α_{\max} should not be too large (nodes will use RTS/CTS mechanism very aggressively otherwise), nor too small (otherwise the existence of hidden nodes will degrade the system performance greatly in the absence of RTS usage). This heuristic is confirmed in Figure 6(a) where $\alpha_{\max} = 3.0$ yields the highest throughput. The explanation is

nearly the same for the curve of β_{\min} shown in Figure 6(b) where $\beta_{\min} = 1.1$ leads to the best performance. A large β_{\min} value will make P_c drop quickly, which means the usage of RTS/CTS mechanism is almost prevented for each transmission. In this case the aggregate throughput of the system degrades sharply as shown in Figure 6(b).

IV. CONCLUSIONS

In this paper, we propose a unique rate adaptation scheme for IEEE 802.11 WLANs, called PBRA. The kernel of PBRA is the probabilistic-based adaptive usage of RTS/CTS in order to strike the balance between collision suppression and added RTS/CTS overheads. Moreover, PBRA differentiates channel-error-induced frame losses from collision-induced ones and hence is able to make more appropriate rate adjustment decisions. The performance of PBRA is evaluated via in-depth simulation with various network topologies, data frame sizes, and wireless channel models. Simulation results show that PBRA clearly outperforms other testing schemes, including ARF, CARA-like schemes, and RRAA. The performance gain is particularly significant in practical random topology networks with fading wireless channels.

REFERENCES

- [1] Ad Kamerman and Leo Monteban, "WaveLAN-II: a high-performance Wireless LAN for the Unlicensed Band," *Bell Labs Technical Journal*, vol.2, no.3, pp.118-133, Aug. 1997.
- [2] M. Lacage, M. H. Manshaei, and T. Turetli, "IEEE 802.11 Rate Adaptation: A Practical Approach," in *Proc. ACM MSWiM*, 2004.
- [3] S. H. Y. Wong, H. Yang, S. Lu, and V. Bharghavan, "Robust Rate Adaptation for 802.11 Wireless Networks," in *Proc. ACM MobiCom*, 2006.
- [4] J. Kim, S. Kim, S. Choi, and D. Qiao, "CARA: Collision-aware Rate Adaptation for IEEE 802.11 WLANs," in *Proc. IEEE InfoCom*, 2006.
- [5] S. Kim, S. Choi, D. Qiao, and J. Kim, "Enhanced Rate Adaptation Schemes with Collision Awareness," in *IFIP Networking*, 2007.
- [6] J. Bicket, "Bit-rate Selection in Wireless Networks," *MIT Master's Thesis*, 2005.
- [7] B. Sadeghi, V. Kanodia, A. Sabharwal, and E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," in *Proc. ACM MobiCom*, 2002.
- [8] Q. Ni, T. Li, T. Turetli, and Y. Xiao, "Saturation Throughput Analysis of Error-Prone 802.11 Wireless Networks," in *JWCMC Journal*, 2005.
- [9] Daji Qiao and Sunghyun Choi, "Goodput Enhancement of IEEE 802.11a Wireless LAN via Link Adaptation," in *Proc. IEEE ICC*, 2001.
- [10] Intersil, "HFA3861B; Direct Sequence Spread Spectrum Baseband Processor," 2000.
- [11] T. S. Rappaport, *Wireless Communications: Principle and Practice*, Englewood Cliffs, NJ: Prentice-Hall, 1996.
- [12] "Network Simulator - ns-2," <http://www.isi.edu/nsnam/ns/>, online link.