Adaptive Optimization of Rate Adaptation Algorithms in Multi-Rate WLANs

Jaehyuk Choi*, Jongkeun Na*, Kihong Park† and Chong-kwon Kim*
*School of Computer Science and Engineering, Seoul National University, Seoul, Korea 151-744
Email: {jhchoi, jkna, ckim}@popeye.snu.ac.kr
†Dept. of Computer Sciences, Purdue University, West Lafayette, IN 47907, U.S.A
Email: park@cs.purdue.edu

Abstract— **Rate adaptation is one of the basic functionalities in today's 802.11 wireless LANs (WLANs). Although it is primarily designed to cope with the variability of wireless channels and achieve higher system spectral efficiency, its design needs careful consideration of cross-layer dependencies, in particular, link-layer collisions. Most practical rate adaptations focus on the time-varying characteristics of wireless channels, ignoring the impact of link-layer collisions. As a result, they may lose their effectiveness due to unnecessary rate downshift wrongly triggered by the collisions. Some recently proposed rate adaptations use RTS/CTS to suppress the collision effect by differentiating collisions from channel errors. The RTS/CTS handshake, however, incurs significant overhead and is rarely activated in infrastructure WLANs. In this paper, we introduce a new approach for optimizing the operation of rate adaptations by adjusting the rate-increasing and decreasing parameters based on link-layer measurement. To construct the algorithm, we study the impact of rate-increasing and decreasing thresholds on performance and show that dynamic adjustment of thresholds is an effective way to mitigate the collision effect in multi-user environments. Our method does not require additional probing overhead incurred by RTS/CTS exchanges and may be practically deployed without change in firmware. We demonstrate the effectiveness of our solution, comparing with existing approaches through extensive simulations.**

I. Introduction

Rate adaptation has become one of the basic functionalities in today's 802.11 WLANs. The goal of rate adaptation is to maximize the transmission goodput by exploiting the multi-rate capability provided by the IEEE 802.11 physical layer (PHY) [1]. The current 802.11 PHY supports a wide range of transmission rates between 1 and 54 Mbps by employing different sets of modulation and channel coding schemes. For example, IEEE 802.11b supports four data rates 1, 2, 5.5, and 11 Mbps whereas 802.11a/g support eight up to 54 Mbps [1], [2]. The basic idea of rate selection is to estimate the channel condition and adaptively select the best rate out of multiple available transmission rates.

The most widely implemented rate adaptation scheme is *automatic rate fallback* (ARF) [18]. In ARF, two consecutive frame transmission failures—i.e., 802.11 Acknowledgement (ACK) frame is not received—result in rate downshift. Ten consecutive frame transmission successes trigger a rate upshift. Asymmetry in the threshold values injects a measure of conservativeness, reflecting the sensitive dependence of bit errors on SNR [10], [11]. Most chip firmware implement variants of the canonical ARF based on up/down counter mechanism [4], [19], [23]. Although well-intentioned, ARF cannot react quickly to fast channel fluctuation since at least 10 attempts are required to increase the transmission rate. Conversely, it may be considered to be overreactive (i.e., attempt rate-increases too often) if the channel condition varies very slowly. This problem stems from the use of fixed up/down thresholds without consideration of channel variation.

The performance and efficiency of rate adaptation depend on the rate control parameters such as up/down thresholds. For example, fast-fading channels require a small value of up-threshold in order for the rate adaptation to keep up with the channel variations [9]. Conversely, for slowly changing channels, the use of a large value of up-threshold can prevent excessive rate-increasing attempts. Several research efforts [9], [20], [25] have dealt with time-varying wireless channel characteristics through adaptive up/down-thresholds. Chevillat et al. [9] proposed to adaptively use a small value and large value of up-threshold to deal with fast and slow fading channels. Qiao et al. [25] proposed a similar approach called *fast-responsive link adaptation* which controls the sender's rate-increasing attempts dynamically to improve responsiveness to the channel variation. Adaptive Auto Rate Fallback (AARF) [20] aims to improve the performance of ARF in slow-fading channels. AARF doubles its up-threshold every time when it tries to increase the transmission rate and the subsequent packet transmission fails.

Most practical rate adaptations focus on the time-varying characteristics of wireless channels, ignoring the impact of link-layer collisions. As a result, they may respond to frame collisions—which cannot be distinguished from channel errors based on missing 802.11 ACKs alone—resulting in unnecessary rate downshift even when channel noise is low. This can significantly decrease throughput when transmission failures are caused by collisions [10], [11], [19].

To suppress the collision effect, some recently proposed rate adaptations [15], [19], [28] leverage the per-frame RTS option and selectively turn on RTS/CTS exchange to differentiate collisions (indicated by a failure of RTS frame) from channel errors (indicated by an unsuccessful data frame transmission following a successful RTS/CTS handshake). However, RTS/CTS is rarely turned on in practical infrastructure IEEE 802.11 WLANs due to high overhead. Per-frame selective RTS will remain a costly solution in lossy environments−especially it becomes a relatively large overhead for short data packets.

In this paper, we consider the problem of mitigating the performance degradation of rate adaptation stemming from link-layer collisions. Our main objective is to find a solution that does not require additional probing overhead such as those incurred by RTS/CTS exchanges. The key idea is that dynamic adjustment of up/down-thresholds (adaptive use of probing interval) can be useful not only to cope with channel dynamics [9], [25] but also to mitigate the impact of collisions. As the number of contending stations increases, the number of collisions is also likely to increase triggering unnecessary—in fact, detrimental—rate-downshifts. In such a situation, a higher value of down-threshold can reduce the undesired rate-decreasing probability. Similarly, a smaller value of up-threshold can help recover from unintended rate-decreases induced by collisions.

Motivated by the above observation, we study the impact of up/down-thresholds on the system performance by utilizing our previously proposed analytical model of ARF [10]. We derive target thresholds that can offset the detrimental collision effect under fixed up/down thresholds. We propose a run-time adaptive algorithm to optimize the operation of rate adaptations by dynamically controlling the rate control thresholds based on link-layer measurements. The simulation results show that the adaptively tuned thresholds are effective not only at offsetting the collision effect but also improving the responsiveness to channel variation. Our solution does not require additional probing overhead and can be practically deployed without change in firmware.

The remainder of the paper is organized as follows. In section II, we formulate the problem and introduce the framework of our approach. Section III analyzes the impact of rate-control parameters on system performance. In Section IV, we describe the proposed scheme and we evaluate the performance of our solution using simulation in Section V. We conclude with a discussion of related work.

II. Problem Formulation

We consider a station in a multi-rate IEEE 802.11 WLAN. Let A denote the station's rate adaptation algorithm which uses two thresholds θ_u^A (up-threshold) and θ_d^A (down-threshold), where θ_u^A consecutive successes trigger a rate upshift (more precisely, up-rate probing [18] [10]) and θ_d^A consecutive transmission failures result in a rate downshift. Note that θ_u^A and θ_d^A can be fixed or variable depending on rate adaptation algorithm A. For example, the canonical ARF uses fixed values of $\theta_u^{ARF} = 10$ and $\theta_d^{ARF} = 2$. AARF [20] uses a binary exponential up-threshold θ_u^{AARF} while its down-

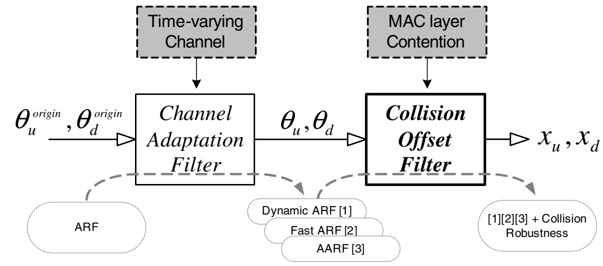


Fig. 1. The framework of our approach—*collision offset filter*

threshold θ_d^{AARF} is fixed to 2 (= θ_d^{ARF}) where

$$\theta_u^{AARF} = \begin{cases} \min(2 \cdot \theta_u^{AARF},\ 50) & \text{if up-rate probing fails,} \\ 10\ (= \theta_u^{ARF}) & \text{if rate-downshift occurs.} \end{cases}$$

Other variants [9] [25] of the canonical ARF also use adaptive thresholds and can be expressed similarly. We note that adaptive thresholds used in these schemes do not consider the collision effect.

Our aim is to mitigate the unintended rate adaptation stemming from collisions. Instead of RTS/CTS, we optimize the operation of rate adaptation A by adjusting its rate-control thresholds based on estimation of link-layer conditions. We formulate the problem as follows

$$x_u^A = f_u(\theta_u^A, N) \text{ and } x_d^A = f_d(\theta_d^A, N) \quad (1)$$

where N represents the link-layer contention status, i.e., number of contending stations. (x_u^A, x_d^A) are adaptively tuned thresholds of (θ_u^A, θ_d^A), aiming to offset the collision effect experienced under operation with (θ_u^A, θ_d^A). Fig. 1 illustrates the overall structure of the proposed adaptive threshold scheme. In short, our main objective is to derive the threshold tuning functions $f_u(\cdot)$ and $f_d(\cdot)$ that are required to create the *Collision Offset Filter*.

The first challenge in deriving functions $f_u(\cdot)$ and $f_d(\cdot)$ is the lack of a *target reference point* (or *target value*) for up/down thresholds that indicates what rate adaptation behavior is optimal to mitigate the collision effect. This is addressed in the next section.

III. Performance of ARF and Its Ideal Behavior

In our previous work [10], we have analytically derived the behavior of ARF and its performance when ARF is in a certain stationary channel condition. In this section we first review the ARF model briefly. Based on its results, we study the impact of up/down thresholds on performance, from which we obtain a key implication to avoid the malfunction due to collisions.

A. Analytic Model of ARF

The analysis considers a station adopting ARF in a multi-rate IEEE 802.11 WLAN with L data rates $R_1 < R_2 < \cdots < R_L$, where the WLAN consists of N stations which are non-cooperative with one another. Let θ_u and θ_d denote the up and down thresholds of ARF, respectively. For each rate R_i and given a fixed frame size, the station is supposed to have a frame error rate (FER) e_i obeying $e_1 \leq e_2 \leq \cdots \leq e_L$ due to the increased robustness of 802.11 PHY modulation at lower data rates. Following Bianchi [6], we introduce the independence assumption that in equilibrium a frame transmission experiences collisions with constant and independent probability p. Thus the conditional transmission failure probability of a frame transmitted at rate R_i is given by $p_i = 1 - (1-p)(1-e_i)$. Note that even though the transmission failure probability p_i consists of p and e_i, ARF can not recognize p and e_i separately and it only behaves according to the value of p_i.

The key observation we can find in the ARF algorithm is that *the transmission rate is always switched to adjacent one*, so that the rate adaptation procedure of ARF could be expressed via *a simple birth-death Markov chain* as shown in Fig. 2, where the state i represents the transmission rate R_i of the single target station. Note that each state in this chain is a macro-state which contains micro-states representing the consecutive counters of ARF (the detail is described in [10]).

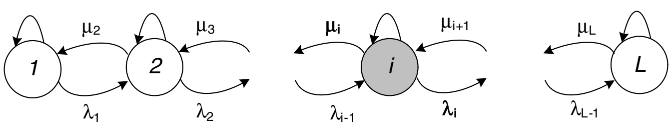


Fig. 2. Birth-death Markov Chain for ARF (L PHY rates)

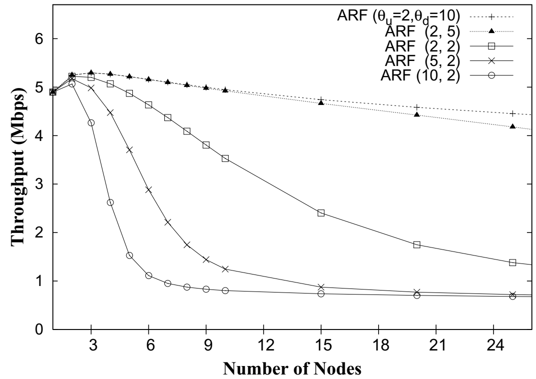
Let Π_i denote the steady-state probability of the ARF chain that captures a station's probability of transmitting at data rate R_i. λ_i ($i \in \{1, 2, \ldots, L-1\}$) and μ_i ($i \in \{2, \ldots, L\}$) denote the state transition rates of increasing the current rate i to $i+1$ and decreasing the current rate i to $i-1$, respectively. Since the equilibrium distribution of a L-state birth-death chain with birth rates λ_i and death rates μ_i is given by

$$\Pi_1 = \frac{1}{1 + \sum_{j=1}^{L-1}(\prod_{k=1}^{j} \frac{\lambda_k}{\mu_{k+1}})} \text{ and } \Pi_i = \frac{\lambda_{i-1}}{\mu_i}\Pi_{i-1}, \quad (2)$$

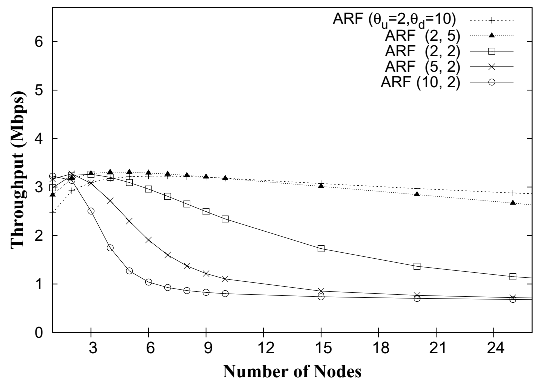
for $i \in \{2, \ldots, L\}$. In [10], we already derived λ_i and μ_i for a stationary and independent p_i and two thresholds θ_u, θ_d as follows:

$$\lambda_i = \frac{p_i(1-p_i)^{\theta_u}}{1-(1-p_i)^{\theta_u}}, \quad \mu_i = p_i^{\theta_d}, \quad (3)$$

which means that when ARF is in a certain stationary channel condition with a transmission failure probability p_i, it increases the current rate i to $i+1$ with the probability of λ_i and decreases the current rate i to $i-1$ with the probability of μ_i, respectively. Eq. (3) also implies that *the rate-shifting probability can be controlled by adjusting its thresholds* θ_u *and* θ_d. It is of great practical importance to understand the behavior of ARF and improve its performance.



(a) SINR=13dB



(b) SINR=9dB

Fig. 3. ARF-DCF throughput for various θ_u and θ_d combinations (a) SINR=13dB (b) SINR=9dB (1000 bytes)

Using the ARF analysis model, we now characterize the impact of both link-layer contention and up/down thresholds on ARF performance. Fig. 3 shows ARF-DCF throughput in 802.11b PHY environment for different combinations of the up/down thresholds as the number of contending station N is varied. We consider two different stationary channel state : (i) moderate noise channel at which SINR=13dB, (ii) high noise at which SINR=9dB, where we use empirical BER versus SNR curves provided by Intersil [3]. All stations are supposed to use equal up/down thresholds.

From the result, we can see the impact of both the link-layer contention and the up/down thresholds on ARF performance. When the number of stations N is small (N=1 or 2), the default values θ_u=10 and θ_d=2 implemented in WLAN cards achieves reasonable performance at both low and high noisy channel (note that the result is for stationary channel, i.e., no fading). However, its performance drops precipitously as the number of contending station N increases. The steep decline in throughput is caused by ARF's inability to effectively differentiate channel noise from collision. When asymmetry is reversed, i.e., θ_u=2 and θ_d=10, it achieves the best performance as N increases unlike it performs worst at SINR=9dB and N=1. Note that this result does not mean that the configuration of (θ_u=2, θ_d=10) is always proper than others since it is likely to suffer seriously under fast-fading environment due to its large down threshold θ_d.

B. Ideal Behavior of ARF

As described in the previous section, it is well-known that when a WLAN has a number of active stations, frequent collisions may happen and ARF loses its effectiveness due to the detrimental rate down-shift operations misbehaved by the collisions. To remedy this ineffectiveness, *ARF should not react to collisions but respond only to channel errors*, i.e. the frame losses due to collisions should be filtered out from ARF's failure counting decision.

In the ideal case, the station has the perfect knowledge of the cause of transmission failures, i.e., whether due to channel errors or collisions, without additional probing expenses such as RTS/CTS exchange. Then, the rate adaptation can perfectly prevent the malfunction due to collisions and hence attain its maximum achievable throughput. In this paper, we define *ideal ARF* (or *Ideal Collision Filtering ARF*) as such ARF having a perfect collision filtering ability without any additional expense. Even though *ideal ARF* is not realizable, we can analytically characterize its behavior and derive performance by using our analysis model.

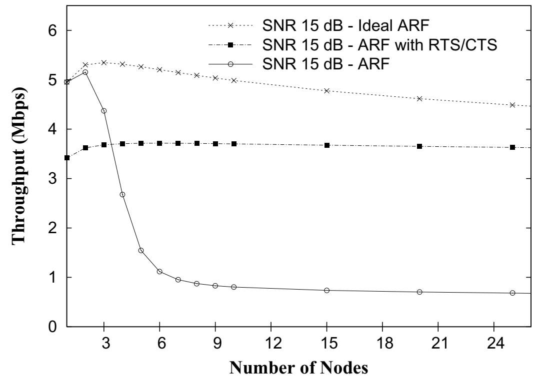
Let $\Pi_i^{opt}(\theta_u, \theta_d)$ denote the probability of transmitting at rate R_i of *ideal ARF* with fixed up/down thresholds θ_u and θ_d. Since *ideal ARF* reacts only to channel errors, its responsive probability to frame errors is not p_i but $(1-p)e_i$ (or $p_i - p$). Therefore, its transition probabilities $\lambda_i^{opt}, \mu_i^{opt}$ at R_i are derived as

$$\begin{aligned} \lambda_i^{opt} &= \frac{(1-p)e_i\{1-(1-p)e_i\}^{\theta_u}}{1-\{1-(1-p)e_i\}^{\theta_u}}, \\ \mu_i^{opt} &= \{(1-p)e_i\}^{\theta_d} \end{aligned} \tag{4}$$

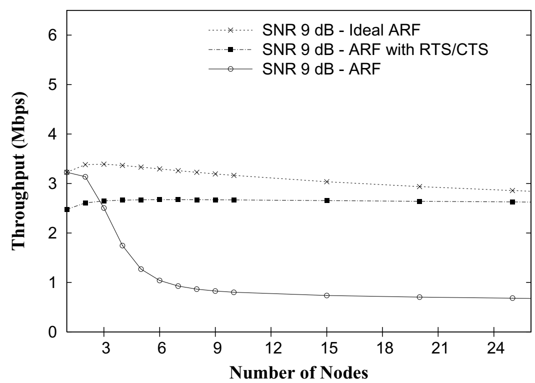
which are obtained by substituting $(p_i - p)$ instead of p_i into Eq. (3). Note that $\lambda_i^{opt} \geq \lambda_i$ and $\mu_i^{opt} \leq \mu_i$ where $\lambda_i^{opt} = \lambda_i$ and $\mu_i^{opt} = \mu_i$ if $p = 0$.

Accordingly, we can obtain the probabilities $\Pi_i^{opt}(\theta_u, \theta_d)$ ($i \in \{1, \ldots, L\}$) using Eq. (2). In Fig. 4, we compare the throughput of original ARF without RTS/CTS, ARF with RTS/CTS, and *ideal ARF* for θ_u=10 and θ_d=2 as an example.

Recall that Eq. (4) characterizes the optimal behavior of a rate adaptation that alleviates the collision effect. Motivated by this result, we thus use λ_i^{opt} and μ_i^{opt} as the *target reference value* to control up/down-thresholds in our algorithm.



(a) SINR=13dB



(b) SINR=9dB

Fig. 4. Performance of *ideal ARF* (θ_u=10, θ_d=2) at (a) SINR=13dB (b) SINR=9dB (1000 bytes)

IV. Optimization of Rate Adaptations : Link-layer Adaptive Threshold

In the previous section, we have analytically studied the impact of rate-increasing and decreasing thresholds on the throughput performance and characterized the ideal behavior that perfectly mitigates the collision effect. Our objective in this section is to find the optimal thresholds (x_u, x_d) offsetting the collision effect experienced when working with the original fixed thresholds (θ_u, θ_d).

A. Basic Idea

When ARF(θ_u,θ_d) experiences a stationary and independent transmission failure probability p_i (following [6]), its rate-shifting probabilities λ_i, μ_i are calculated as in Eq. (3) and its ideal ones λ_i^{opt}, μ_i^{opt} correspond to Eq. (4). The ideal rate-shifting probabilities are obtained under the assumption that ARF is able to differentiate collisions from channel errors perfectly.

The basic idea of our approach is simple. Considering λ_i^{opt}, μ_i^{opt} as the *target values*, we make λ_i and μ_i at rate R_i converge to λ_i^{opt} and μ_i^{opt} by adaptively adjusting the thresholds θ_u, θ_d. This implies that the rate distribution Π_i at rate R_i ($i \in \{1, 2, \ldots, L\}$) will asymptotically converge to optimal Π_i^{opt}.

To formulate our approach, we express the rate-shifting probabilities λ_i and μ_i in Eq. (3) as $\lambda_i(\theta_u, p_i)$ and $\mu_i(\theta_d, p_i)$. Similarly, let us represent the rate-shifting probabilities λ_i^{opt} and μ_i^{opt} in Eq. (4) as $\lambda_i^{opt}(\theta_u, p, e_i)$ and $\mu_i^{opt}(\theta_u, p, e_i)$, respectively. The collision mitigating thresholds x_u, x_d are obtained by satisfying the followings:

$$\lambda_i(x_u, p_i) = \lambda_i^{opt}(\theta_u, p, e_i),$$
$$\mu_i(x_d, p_i) = \mu_i^{opt}(\theta_d, p, e_i), \quad (5)$$

which yield

$$x_u = \frac{\ln \frac{\lambda_i}{\lambda_i + p_i}}{\ln(1-p_i)} = \frac{\ln \dfrac{(1-p)e_i(1-(1-p)e_i)^{\theta_u}}{p_i + p(1-(1-p)e_i)^{\theta_u}}}{\ln(1-p_i)},$$
$$x_d = \frac{\ln \mu_i}{\ln p_i} = \theta_d \cdot \frac{\ln(1-p)e_i}{\ln p_i}, \quad (6)$$

where $p_i = 1-(1-p)(1-e_i)$. Then, $[x_u]$ and $[x_d]$ are used as the new thresholds where $[x]$ represents the integer closest to x.

If we know the collision probability p and the frame error probability e_i ideally, we can obtain the link-layer adaptive thresholds x_u, x_d easily according to Eq. (6). However, this requires for the stations knowing or estimating e_i for each rate ($i \in \{1, 2, \ldots, L-1\}$) and p separately. Note that ARF is not based on statistical estimation, thus it does neither estimate nor use the transmission failure rate p_i. In our approach, we also avoid the estimation of channel condition e_i (virtually, it is difficult to acquire and predict the instant channel error rate accurately without the modification of current 802.11 standard). Instead, our scheme only makes use of a link-layer measurement through following observation; even though stations in a 802.11 WLANs cannot differentiate collisions from channel errors for each transmission failure, the stations can estimate the link-layer status (i.e. the collision probability p or the number of competing stations N) by using many existing practical on-line measurement and estimation algorithms [7], [17], [21], [26] (in the following section, we will discuss the detailed estimation method for the collision probability p).

In short, to be practically implemented as a control algorithm, we aim to obtain the algebraic functions $f_u(\cdot)$ and $f_d(\cdot)$ independent to channel error term e_i as follows:

$$x_u^A = f_u(\theta_u^A, N) \text{ and } x_d^A = f_d(\theta_d^A, N),$$

which require only a link-layer measurement.

B. Adaptive Threshold independent to Channel Condition

We express x_u, x_d in Eq. (6) as $x_u = f'_u(\theta_u, p, e_i)$, $x_d = f'_d(\theta_d, p, e_i)$. To build the algorithm not requiring channel information such as Eq. (1), the design should remove the e_i term from the functions $f'_u(\theta_u, p, e_i)$ and $f'_d(\theta_d, p, e_i)$.

For a given collision probability p, the adaptive thresholds x_u, x_d have different values according to channel error e_i. Fig. 5 plots the $f'_u(\theta_u, p, e_i)$ function for several values of collision probability p with respect to $\forall e_i$ ($0 < e_i \leq 1$), i.e. $p < p_i \leq 1$, where we set the rate-increasing threshold θ_u to 10.

From Fig. 5, we can see that the range of $f'_u(\cdot)$ (i.e. x_u) for various e_i is not so broad except for the large values of e_i (whose resultant $p_i \approx 1$). Here, a notable observation is that

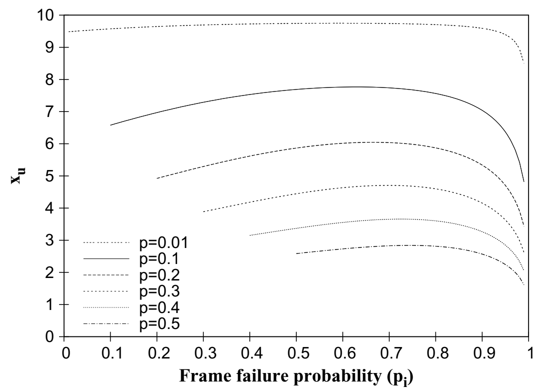


Fig. 5. $x_u = f'_u(\theta_u, p_i)$ for arbitrary channel condition ($p < p_i \leq 1$) for θ_u = 10

the conservative nature of rate adaptations keeps the channel condition at the low noise regime (i.e. rate adaptations select a transmission rate at which the channel noise is low, namely e_i is almost always maintained small). We can thus ignore the high noise region (large e_i) in Fig. 5. Then, since the range of x_u for the valid range of e_i becomes narrow, we can thus use a integer closest to x_u for $p < p_i \ll 1$ as the final value of $f'_u(\cdot)$. In our design, to simplify the algorithm and avoid excessive control, we use a conservative heuristic that sets $x_u = \max\{f_u(\theta_u, p, e_i)\}$ for $\forall e_i$ ($0 < e_i \leq 1$). For example, we have chosen $x_u = 4.7$ for $p = 0.3$ in Fig. 5. Similarly, we set $x_d = \min\{f_d(\theta_d, p, e_i)\}$ for $\forall e_i$ ($0 < e_i \leq 1$). Note that the smaller value of x_u and larger value of x_d imply being in more aggressive control. This is the rationale of the conservative setting.

In the result, we obtain the control function in Eq. (1) for rate adaptation A with thresholds (θ_u, θ_d) as follows:

$$x_u = f_u(\theta_u^A, p) = \max_{p<p_i\leq 1}\left\{\frac{\ln \dfrac{(p_i-p)(1-(p_i-p))^{\theta_u}}{p_i + p(1-(p_i-p))^{\theta_u}}}{\ln(1-p_i)}\right\},$$
$$x_d = f_d(\theta_d^A, p) = \min_{p<p_i\leq 1}\left\{\theta_d \cdot \frac{\ln(p_i-p)}{\ln p_i}\right\}. \quad (7)$$

For example, we obtain the link-layer adaptive thresholds x_u, x_d for ARF(θ_u=10, θ_d=2) with respect to the number of contending stations N and its resultant collision probability p [6] in Table. I.

Consider an example of $N = 5$ whose collision probability is $p = 0.181$. For ARF working with default thresholds θ_u^{ARF}=10 and θ_d^{ARF}=2, its adaptive thresholds are x_u = f_u(10, 5) = 6.34 and x_d = f_d(2, 5) = 3.29 according to Table. I. Since thresholds should be integers, we use $[x_d] = 6$, $[x_u] = 3$ approximately. Fig. 6 compares the throughput (analytical result) under $N = 5 (p \approx 0.18)$ for different combinations of up/down thresholds over a wide range of channel conditions. Fig. 6 shows that for $N = 5 (p = 0.18)$, our adaptive method (x_u=6, x_d=3) offsets the collision effect experienced

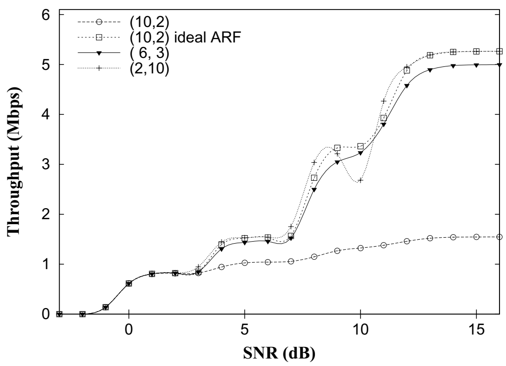


Fig. 6. ARF-DCF throughput for various θ_u, θ_d combinations under $N = 5$

under working with (θ_u^{ARF}=10, θ_d^{ARF}=2). In the figure, we also compare our result with more aggressive control (x_u=2, x_d=10) which have performed the best in Fig. 3. The collision effect is almost mitigated with (x_u=2, x_d=10) but it does not work properly for a certain range of channel errors (near 10dB) − moreover, it may not react quickly to fast-fading channel due to its large down threshold.

Finally, we need a control algorithm to estimate the link-layer collision probability p (or number of contending station N) and to make thresholds x_u, x_d converge to the target values. In the following section, we discuss the link-layer estimation and propose a run-time control algorithm.

C. Estimation of Link-layer Condition

In a WLAN, the collision probability p is a common shared variable to all the contending stations and possible to be estimated by each individual station. As proposed in [7], p can be measured directly by monitoring channel activity and counting the number of experienced collisions and the number of observed busy slots. However, this approach has a difficulty to acquire p accurately due to its complexity. As an alternative state variable inferring channel condition, several practice on-line measurement algorithms [14], [17] use the number of consecutive idle slots between two busy periods (denoted by n_{Idle}). In our scheme, we also use n_{Idle} as a reference value to estimate the collision probability p and the number of competing stations N. However, most existing works for estimating the number of contending stations only consider the non-channel error environment even though there is strong dependency between n_{Idle} and channel errors. First, therefore, we verify that n_{Idle} could be a reliable reference value even in the case of existence of channel errors.

TABLE I
VALUES OF (x_u, x_d) FOR ARF (θ_u=10, θ_d=2)

N	p	x_u	x_d	N	p	x_u	x_d
1	0	10	2	11	0.308	4.61	4.74
2	0.059	8.62	2.35	12	0.322	4.45	4.94
3	0.107	7.63	2.68	13	0.335	4.31	5.14
4	0.147	6.90	2.99	14	0.346	4.19	5.32
5	0.181	6.34	3.29	15	0.357	4.08	5.50
6	0.210	5.90	3.57	20	0.402	3.64	6.33
7	0.235	5.54	3.83	25	0.436	3.34	7.08
8	0.256	5.25	4.07	30	0.463	3.12	7.75
9	0.276	5.00	4.31	40	0.507	2.79	9.03
10	0.293	4.79	4.53	50	0.540	2.57	10.19

1) Impact of Channel Errors on Idle periods: It is well known that n_{Idle} is geometrically distributed with parameter $1-P_I$, where P_I is the probability that a slot remains idle, i.e. $P_I = (1-\tau)^N$ [14], [17] and τ denotes the attempt rate of a station [6]. Thus, the average number of consecutive idle slots is calculated as $E[n_{Idle}] = \frac{(1-\tau)^N}{1-(1-\tau)^N}$, and N can be derived as

$$N = \frac{log(\frac{E[n_{Idle}]}{E[n_{Idle}]+1})}{log(1-\tau)}, \quad (8)$$

In the case of existence of channel errors, the attempt rate τ changes (i.e. τ decreases as channel noise increases due to binary exponential backoff rule [8]), and so do the estimated value of N according to Eq. (8) while the number of actual contending stations is stationary. Fig. 7 shows the impact of channel errors on the uniqueness of relation between n_{Idle} and N. Fig. 7 implies that N can not be obtained uniquely based on the estimated n_{Idle} since the same value of n_{Idle} indicates diverse N according to the channel errors.

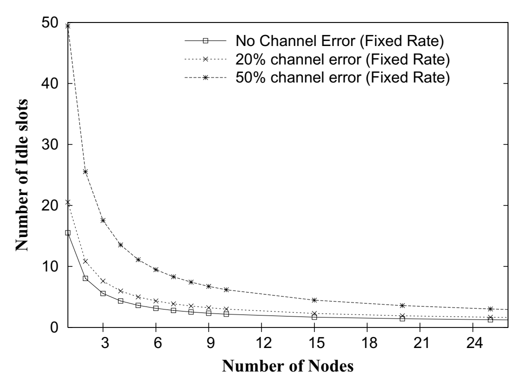


Fig. 7. The impact of channel errors on n_{Idle} with respect to the number of contending stations (homogeneous environment)

2) ARF's Regulation Effect on Idle periods: ARF is very sensitive to frame losses and selects a conservative data rate at which the transmission error probability e_i is low. That implies that ARF maintains the channel condition at the low noise regime. Fig. 8 shows the impact of ARF on n_{Idle} reflecting N for various channel conditions. Unlike Fig. 7, the result shows that n_{Idle} is a good reference state value to estimate N regardless of channel state (SNR) due to the ARF's (i.e. rate adaptation's) conservative rate selection property.

TABLE II

OFFLINE ADAPTIVE THRESHOLD TABLE FOR E[n_{Idle}] (θ_u=10, θ_d=2) : $x_u = f_u(10, \text{E}[n_{Idle}]), x_d = f_d(2, \text{E}[n_{Idle}])$

$E[n_{Idle}]$	(p)	x_u	$E[n_{Idle}]$	(p)	x_d
11.64−	(<0.02)	10	>6.53	(<0.08)	2
7.55−11.64	(0.02−0.06)	9	3.16−6.53	(0.08−0.20)	3
5.21−7.55	(0.06−0.11)	8	2.15−3.16	(0.20−0.29)	4
3.72−5.21	(0.11−0.17)	7	1.66−2.15	(0.29−0.36)	5
2.70−3.72	(0.17−0.24)	6	1.35−1.66	(0.36−0.41)	6
1.93−2.70	(0.24−0.32)	5	1.15−1.35	(0.41−0.45)	7
1.31−1.93	(0.32−0.42)	4	1.00−1.15	(0.45−0.49)	8
0.79−1.31	(0.42−0.55)	3	0.89−1.00	(0.49−0.52)	9
0.31−0.79	(0.55−0.76)	2	0.80−0.89	(0.52−0.55)	10
−0.31	(>0.76)	1	<0.80	(>0.55)	11

D. Control Algorithm

By observing the channel state, we can estimate the number of consecutive idle slots n_{Idle}. To provide a run-time adaptive estimation reflecting the network dynamics, the estimation is updated using the moving average, i.e.

$$E[n_{Idle}]_i \leftarrow (1-\alpha) \cdot E[n_{Idle}]_{i-1} + \alpha \cdot (n_{Idle})_i. \quad (9)$$

In our approach, to avoid the complexity, we do not calculate the collision probability p or number of contending station N explicitly based on the estimated $E[n_{Idle}]$. Instead, at offline, we establish the threshold tuning tables indexed by θ_u (and θ_d) and $E[n_{Idle}]$, i.e. $f_u(\theta_u, E[n_{Idle]}), f_d(\theta_d, E[n_{Idle]})$. Therefore, we can obtain adaptive thresholds $x_{u_{new}}$ and $x_{d_{new}}$ by simple run-time table lookup based only on $E[n_{Idle}]$. For example, Table. II is established for the initial thresholds (x_u=10, x_d=2) at offline.

Then, the currently used thresholds x_u and x_d are also updated using the moving average as follows:

$$\begin{aligned} x_u &\leftarrow (1-\beta) \cdot x_u + \beta \cdot x_{u_{new}}, \\ x_d &\leftarrow (1-\beta) \cdot x_d + \beta \cdot x_{d_{new}}. \end{aligned} \quad (10)$$

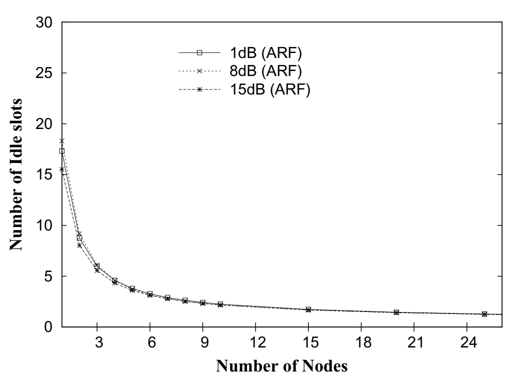


Fig. 8. The impact of conservative nature of ARF on the relation between N and n_{Idle} in various channel environments

V. PERFORMANCE EVALUATION

A. Simulation Setup

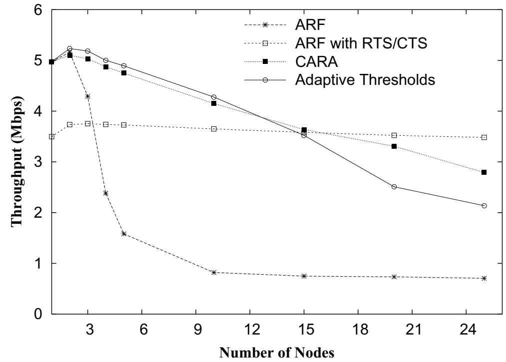
In this section, we evaluate our proposed scheme by simulations in *ns-2* simulator [5]. We have implemented our scheme in *ns-2* version 2.31. For comparison, we also implemented ARF [18] and CARA [19]. All simulations are performed in an infrastructure-based WLAN environment, with one destination station and other source stations. We consider the IEEE 802.11b PHY specification, thus a node can choose one of the four data rates, 1, 2, 5.5, and 11Mbps. Data traffic is generated using constant bit rate (CBR) UDP traffic sources, and simulations are performed in saturated conditions, i.e., there is more traffic than the network can accommodate. All nodes have no mobility. The moving average coefficients in Eqs. (9), (10) are set to α=0.1 and β=0.5.

B. Stationary Channel Condition

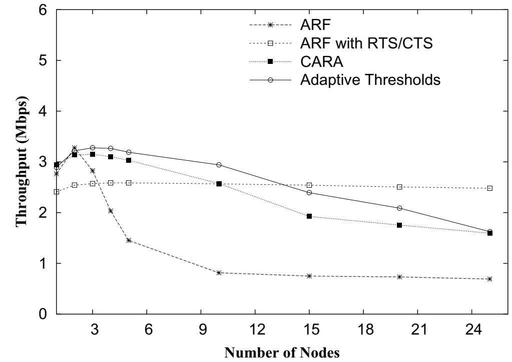
To illustrate the effectiveness of our proposal, we first have performed simulations in stationary channel environments. We compare the following schemes: (1) ARF [18], (2) ARF using the RTS/CTS exchange all the time (referred as to ARF+RTS), (3) CARA [19], and (4) our proposed link-layer adaptive scheme. The test schemes are compared with each other in terms of the aggregate system throughput (in Mbps). As addressed in Section IV, we set the consecutive success threshold (θ_u) to 10, and the consecutive failure threshold (θ_d) to 2, for ARF and CARA. We use the empirical BER (Bit Error Rate) vs. SNR (Signal-to-Noise Ratio) curves, provided by Intersil [3], to set the FER (Frame Error Rate). The RTS/CTS frames are always transmitted at the lowest rate of 1 Mbps. We conduct the simulations under various channel states and data frame size.

Fig. 9 presents the throughput performance of ARF, ARF+RTS, CARA, and our scheme as the number of stations increases from 1 to 25 in 802.11b channel. The throughput of ARF suffers as the number of stations increases. We can see that the cause of significant performance degradation (a bell shaped throughput curve) of ARF is that ARF cannot differentiate collisions from channel errors. On the other hand, even as multiple access contention increases from 1 to 25, the throughput of ARF+RTS remains flat, which implies that ARF+RTS works properly even with many contending stations. It is because the RTS/CTS exchange plays a role that filters out collisions from channel errors.

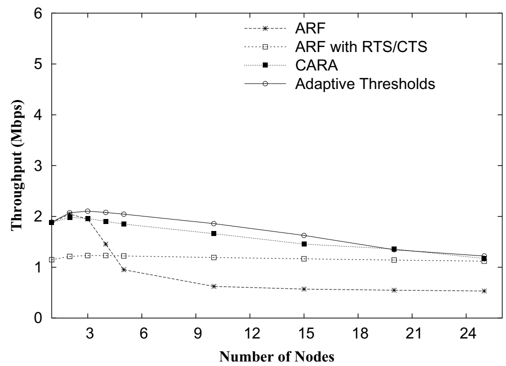
The result also shows that our proposed adaptive threshold scheme offsets the collision effect of ARF fairly. When the number of stations (N<10) is small−which is the mainly used realistic settings of today's WLANs− with 1000 bytes payload (Fig. 9(a) and (b)), the performance of our scheme is shown to be better than others. For the case of high multiple access contention(N>10), the schemes using RTS/CTS handshake performs better than our scheme. However, its source stems from the less waste time of RTS frame than data frames when



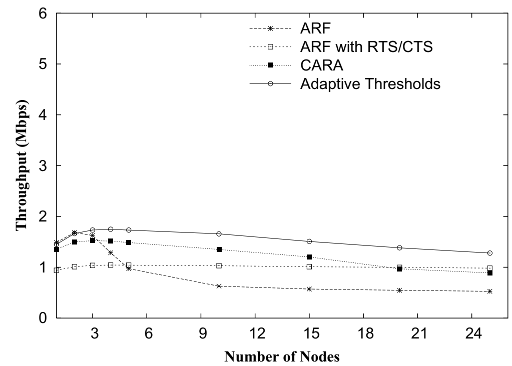
(a) 1000 bytes, SINR = 15dB



(b) 1000 bytes, SINR = 10dB



(c) 250 bytes, SINR = 15dB



(d) 250 bytes, SINR = 10dB

Fig. 9. Throughput comparison of our proposal (Adaptive Thresholds) against ARF, ARF with RTS/CTS, and CARA in stationary channel condition at which (a) SINR=15dB, 1000 bytes (b) SINR=10dB, 1000 bytes (c) SINR=15dB, 250 bytes (d) SINR=10dB, 250 bytes

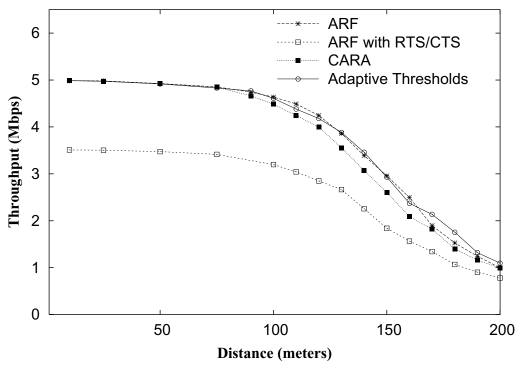
a collision occurs, which is the MAC layer issue, i.e. it is not the focus of this paper.

In the real environments, however, our scheme is expected to be utilized satisfactorily related to the real distribution of Internet packet size. According to the report from Cooperative Association for Internet Data Analysis (CAIDA) [22], the actual Internet traffic is governed by small sized packets of under 100 bytes with more than 50%, and another peak is at 1500 bytes corresponding to TCP's maximum transfer unit (MTU). Therefore, the RTS/CTS handshake becomes a relatively large overhead for the small packets. We examine the performance for small packets (250 bytes payload) in Figs. 9(c) and (d). The result shows that the RTS/CTS overhead reduces the overall throughput at which our scheme performs better than others.

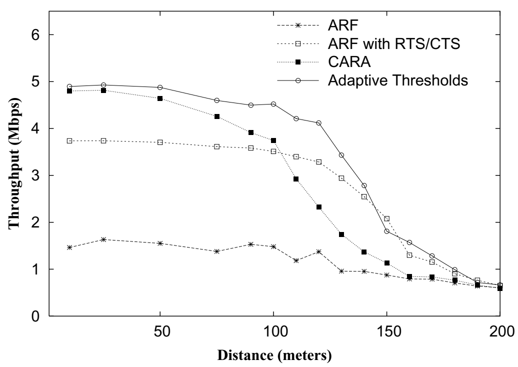
C. Fading Channel Environment

We now consider the multi-path fading effect with which the channel condition varies over time. We first use Ricean fading as the propagation model to simulate the time-varying wireless channel condition. Fig. 10 compares the throughput of the testing schemes as a function of the distance for (a)N=1 and (b)N=5. We see that when the number of station is one (Fig. 10(a)), the performance of ARF and our scheme is almost same since our scheme uses the identical thresholds with ARF (note that no contention in this case). The CARA also performs close to both ARF and ours but a bit lower due to overhead of (selective) RTS/CTS exchange. The ARF+RTS scheme performs worse than others due to the overhead of RTS/CTS frame exchanges before each data transmission attempt. For the case of $N = 5$ (Fig. 10(b)), we can see that the performance of ARF significantly decreases due to increased contention. The result shows that our scheme significantly improves the performance of ARF and performs the best among the concerned schemes. The improvement is achieved thanks to the small value of up-threshold x_u which enables our scheme to react to time-varying channel quickly. This result implies that the adaptive adjustment of the thresholds helps not only mitigate the collision effect but also improve the responsiveness to the channel variation.

Next, we examine the performance of our scheme under various fading channel condition using Rayleigh fading model (via different Doppler spread values–larger value represents more fast channel). The distance between AP and stations is fixed to 200m and the number of stations is N=5 in this scenario. Fig. 11 shows the simulation results with different Doppler spread values. Although the throughput of all schemes decreases with increasing Doppler spread values, our scheme outperforms others. Similar to the result in Fig. 10, this is because the dynamic adjustment of threshold improves the responsiveness to the channel variation.



(a) N=1



(b) N=5

Fig. 10. Throughput comparison in Ricean fading channel (payload = 1000 bytes) for N=1 and 5

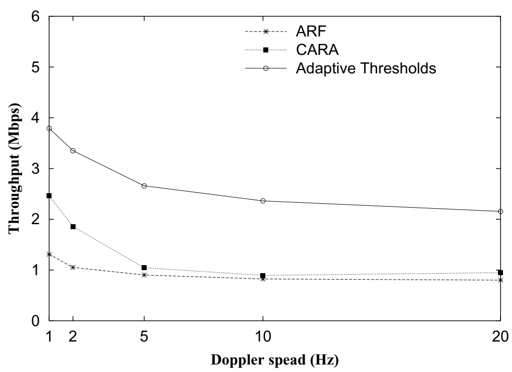


Fig. 11. Throughput comparison under various fading channel (Rayleigh fading) (payload = 1000 bytes) for N=5

VI. Related Work

In recent years, rate adaptation has been an active research topic and a number of algorithms [9], [16], [19], [20], [23], [25], [27], [28] have been proposed. Rate adaptation is left to vendors (i.e., is not specified in the IEEE 802.11 standard), yet its design plays a critical role in determining overall system performance [10], [11]. The current 802.11 standard does not provide the receiver's explicit feedback information on the best rate or perceived SNR to the sender. Therefore, most practical rate adaptations are implemented at sender-side, utilizing information gleaned from successful/unsuccessful ACK transmissions.

ARF [18] is the most widely implemented rate adaptation algorithm, which uses a up/down counter mechanism to set the data rate. In the Enterasys RoamAbout 802.11 DS High Rate card (Orinoco chipset), two consecutive frame transmission failures result in rate downshift. Ten consecutive frame transmission successes trigger a rate upshift. Although well-intentioned, the design of ARF has not taken into account possible frame losses due to collisions. It assumes that all frame losses are due to channel errors, so that ARF may respond to frame collisions resulting in unnecessary rate downshifts even when channel noise is low. Empirical 802.11b WLAN measurements [11] show that even under moderate multiple access contention (4–15 wireless stations) WLAN throughput declines drastically, not because of network congestion but ARF confusing collision with channel noise.

ARF has been extended in two directions; first, to cope with the time-varying wireless channel characteristics [9], [20], [24], [25], and second, to deal with ARF's noise vs. collision differentiation problem [15], [19], [23]. An overview of existing methods can be found in [23] and [12]. To deal with the fast-fading and slow-fading wireless channels, the authors of [9] enhanced ARF to adaptively use a short probing interval and a long probing interval. In [25], a novel fast-responsive link adaptation scheme has been proposed, which directs the transmitter stations rate-increase attempts in a controlled manner such that the responsiveness of the link adaptation scheme can be guaranteed with minimum number of rate-increasing attempts. Kim et al. proposed a modified ARF, called *Collision-Aware Rate Adaptation* (CARA), leveraging the per-frame RTS option in [19]. CARA exploits the fact that RTS frames are small and always encoded at the lowest rate. A RTS frame transmission failure is likely the result of collision whereas data frame transmission failures following a successful RTS/CTS handshake are likely due to channel error. CARA shows improved system performance thanks to its collision-awareness capability. The schemes proposed in [15], [28] use RTS/CTS mechanisms similar to CARA.

Whereas most works in ARF have focused on improving performance through enhanced algorithms and protocol mechanisms, our previous work [10] focused on improving understanding of ARF's dynamics. We proposed a new Markov chain model for ARF and provided a rigorous performance analysis of ARF.

VII. Conclusion

In this paper, we have proposed a new approach that mitigates the collision effect on the operation of rate adaptation in IEEE 802.11 WLANs by adaptively adjusting the rate-increasing and decreasing parameters with simple link-layer channel estimation, i.e., the number of consecutive idle slots. We have studied the impact of rate-increasing and decreasing thresholds on performance and shown that dynamic adjustment of thresholds is an effective way to mitigate the collision effect—unintended and detrimental rate downshift caused by collision, not channel noise—in multi-user environments. We have derived the threshold tuning functions $f_u(\cdot)$ and $f_d(\cdot)$ which are exploited to establish the adaptive threshold tables. We have proposed a run-time algorithm that adaptively controls the operating thresholds by simple run-time table lookup based on the measured number of consecutive idle slots. Through *ns-2* simulations, we have demonstrated that the proposed solution effectively offsets the collision effect, yielding significant performance gains compared to using fixed thresholds. The simulation results have also shown that our solution improves responsiveness to channel variation. While we couch our solution in the context of ARF, the approach may be applicable to other sender-based schemes.

VIII. Acknowledgements

This research was supported, in part, by a BK-21 grant and Seoul R&BD Program.

References

[1] IEEE, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. IEEE Std 802.11-1999, 1999.
[2] IEEE 802.11g, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Further Higher Data Rate Extension in the 2.4 GHz Band*, Supplement to IEEE 802.11 Standard, 2003.
[3] Intersil, "HFA3861B; Direct Sequence Spread Spectrum Baseband Processor," 2000.
[4] http://www.enterasys.com/, online link.
[5] "The Network Simulator. ns-2," http://www.isi.edu/nsnam/ns/, online link.
[6] G. Bianchi, "Performance Analysis of the IEEE 802.11 Distributed Coordinated Function," *IEEE J. Selected Areas in Commun.*, vol. 18, no. 3, pp. 535-547, 2000.
[7] G. Bianchi and I. Tinnirello, "Kalman filter Estimation of the Number of Competing Terminals in an IEEE 802.11 Network," in *Proc IEEE INFOCOM'2003*, vol.2, pp. 844-852, 2003.
[8] P. Chatzimisios, A. C. Boucouvalas, and V. Vitsas, "Performance Analysis of IEEE 802.11 DCF in Presence of Transmission Errors," in *Proc. IEEE ICC'2004*, pp. 3854-3858, 2002.
[9] P. Chevillat, J. Jelitto, A. Barreto, and H. Truong, "A Dynamic Link Adaptation Algorithm for IEEE 802.11a Wireless LANs," in *Proc. IEEE ICC03*, pp. 1141-1145, 2003.
[10] J. Choi, K. Park, and C. Kim, "Cross-Layer Analysis of Rate Adaptation, DCF and TCP in Multi-Rate WLANs," in *Appear. IEEE INFOCOM'07*, 2007.
[11] S. Choi, K. Park, and C. Kim, "On the Performance Characteristics of WLANs: Revisited," in *Proc. ACM SIGMETRICS'05*, pp. 97-108, 2005.
[12] I. Hatatcherev, K. Langendoen, R. Lagendijk, H. Sips, "Hybrid rate control for IEEE 802.11," in *Proc. ACM MobiWac'04*, pp. 10-18, 2004.
[13] M. Heusse, F. Rousseu, G. Berger-Sabbatel, and A. Duda, "Performance Anomaly of 802.11b," in *Proc. IEEE INFOCOM'03*, pp. 836-843, 2003.
[14] M. Heusse, F. Rousseu, R. Guilier, and A. Duda, "Idle Sense: An Optimal access Method for high throughput and Fairness in Rate diverse wireless LANs," in *Proc. ACM SIGCOMM'05*, pp. 121-132, 2005.
[15] C. Hoffmann, M. Manshaei, T. Turletti, "CLARA: closed-loop adaptive rate allocation for IEEE 802.11 wireless LANs," in *Proc. IEEE WIRELESSCOM'05*, pp. 668-673, 2005.
[16] G. Holland, N. Vaiday and P. Bahl, "A Rate-Adaptive MAC Protocol for Multi-Hop wireless Networks," in Proc. ACM MobiCom '01, Jul. 2001.
[17] C. Hun, and J. C. Hou, "A Novel Approach to Contention Control in IEEE 802.11e-Operated WLANs," to *Appear. IEEE INFOCOM'07*, 2007.
[18] A. Kamerman and L. Monteban, "WaveLAN 2: A High-performance Wireless LAN for the Unlicensed Band," *Bell Labs Tech. Journal*, vol.2, no. 3 , pp. 118-133, 1997.
[19] J. Kim, S. Kim, S. Choi, and D. Qiao,"CARA: Collision-Aware Rate Adaptation for IEEE 802.11 WLANs,"in*Proc. IEEE INFOCOM'06*, pp. 1-11, 2006.
[20] M. Lacage, M. H. Manshaei, T. Turletti, "IEEE 802.11 rate adaptation: a practical approach," in *Proc. ACM MSWiM'04*, pp. 126-134, 2004.
[21] H. Ma, X. Li, H. Li, P. Zhang, S. Luo, and C. Yuan, "Dynamic Optimization of IEEE 802.11 CSMA/CA Based on the Number of Competing stations," in *Proc IEEE ICC'2004*, pp. 191-195, 2004.
[22] S. McCreary, and K. Claffy, "Trends in wide are IP traffic patterns: A view from Ames Internet Exchange," ITC Specialist Seminar, 2000
[23] Q. Pang, Leung, V. Liew, S. Liew, "A rate adaptation algorithm for IEEE 802.11 WLANs based on MAC-layer loss differentiation," in *Proc. IEEE Broadband Networks'05*, pp. 709-717, 2005.
[24] J.P. Pavon, S. Choi, "Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement," in *IEEE ICC 2003*, pp.1108-1113, 2003.
[25] D. Qiao and S. Choi, "Fast-Responsive Link Adaptation for IEEE 802.11 WLANs," in *Proc. IEEE ICC'05*, pp. 3583-3588, 2005.
[26] D. Qiao and K. Shin, "Achieving efficient channel utilization and weighted fairness for data communications in IEEE 802.11 WLAN under the DCF," in *Proc. of International Workshop on Quality of Service*, pp. 227-236, 2002.
[27] B. Sadeghi, V. Kanodia, A. Sabharwal and E. Knightly, "Opportunistic Media Access for Multirate Ad Hoc Networks," in in *Proc. ACM MobiCom'02*, pp. 24-35, 2002.
[28] H.Y. Wong, H. Yang, S. Lu, and V.Bharghavan, "Robust Rate Adaptation for 802.11 Wireless Networks," in *Proc. ACM Mobicom'06*, pp. 146-157, 2006