



Rate avalanche: Effects on the performance of multi-rate 802.11 wireless networks

Liqiang Zhang^{a,*}, Yu-Jen Cheng^a, Xiaobo Zhou^b

^a Department of Computer & Information Sciences, Indiana University South Bend, South Bend, IN 46615, USA

^b Department of Computer Science, University of Colorado at Colorado Springs, Colorado Springs, CO 80918, USA

ARTICLE INFO

Article history:

Available online 9 September 2008

Keywords:

Wireless local area networks
Rate adaptation
Media access control

ABSTRACT

Originally designed to deal with the hidden node problem, the Request-to-Send/Clear-to-Send (RTS/CTS) exchange is often turned off in most infrastructure-based 802.11 networks with the belief that the benefit it brings might not even be able to pay off its transmission overhead. While this is often true for networks using fixed transmission rate, our investigation leads to the opposite conclusion when multiple transmission rates are exploited in WLANs. Through extensive simulations using realistic channel propagation and reception models, we found out that in a heavily loaded multi-rate WLAN, a situation that we call *rate avalanche* often happens if RTS/CTS is turned off: high collision rates not only lead to retransmissions but also drive the nodes to switch to lower data rates; the retransmissions and the longer channel occupation caused by lower rates will further deteriorate the channel contention, which yields more collisions. This vicious circle could significantly degrade the network performance even no hidden node presents. Our investigation also reveals that, in the absence of effective and practical loss differentiation mechanisms, simply turning on the RTS/CTS could effectively suppress the rate avalanche effect. Various scenarios/conditions are extensively examined to study the impact of RTS/CTS on the network performance. Our study provides some important insights about using the RTS/CTS exchange in multi-rate 802.11 WLANs.

© 2008 Elsevier B.V. All rights reserved.

1. Introduction

With its great success in the last decade, the prevailing IEEE 802.11 wireless local area network (WLAN) technology still attracts significant research efforts today to further enhance its performance. This is driven by both external factors, such as the ever-growing QoS-sensitive multimedia applications, and the inherent internal deficiencies of wireless communications, such as scarce bandwidth resources and time-varying channel conditions. Besides the advances on the physical layer (PHY) techniques for achieving higher spectra efficiency and link reliability such as MIMO communications, many other efforts have been spent on the medium access control (MAC) sublayer, PHY layer, or cross-layers to tune various parameters for the purpose of performance optimization. Dynamically adjusting transmission rates according to the time-varying and location-dependent link quality is one example of such efforts.

Many current and proposed wireless networking standards support multiple transmission rates at the PHY layer that use different modulation and coding schemes. For example, the 802.11a PHY supports eight transmission rates (6–54 Mbps). Recently, a number of rate adaptation schemes have been proposed in the literature [3,5,9–12,15,19,29,30]. These schemes could be roughly divided into two categories: SINR (Signal to Interference plus Noise Ratio)-based [5,10,15,29] and

* Corresponding author. Tel.: +1 574 520 4297.

E-mail address: liqzhang@iusb.edu (L. Zhang).

statistics-based [3,9,11,12,19,30]. The former usually leads to a higher performance gain than the later, because channel conditions are measured more accurately and timely in the former thus optimal rates are more likely to be selected to match the channel conditions in a timely manner. However, several factors have made SINR-based approaches hard to implement in practice (refer to Section 2 for details). On the other hand, statistics-based rate adaptation algorithms, such as ARF (Auto Rate Fallback) [11], still remain the most widely deployed rate control schemes in today's 802.11 networks because of their simplicity. Therefore, we limit our efforts on enhancing the present statistics-based schemes and keeping them compatible with currently deployed 802.11 WLANs, which we believe is of practical importance.

During our study on the performance of rate adaptation mechanisms, we found that in a heavily loaded multi-rate WLAN using ARF, a situation that we call *rate avalanche* often happens if RTS/CTS exchange is not used. That is, high collision rates not only lead to retransmissions but also drive the nodes to switch to lower data rates; the retransmissions and the longer channel occupation caused by lower rates will further deteriorate the channel contention, which yields more collisions. This vicious circle could significantly degrade the network performance even no hidden node presents. There are two important reasons behind this phenomenon: (1) statistics-based rate adaptation schemes, like ARF, lack the ability to differentiate frame losses caused by collisions from those caused by link errors; all the packet losses, even caused by collisions, are counted to reduce the transmission rate; lower transmission rates deteriorate channel contentions which leads to further rate reducing and more nodes involvement in the cycle and (2) even if a node is luckily less involved in the rate reducing cycle, its performance is still dragged down to the same level as those transmitting at lower rates, which is a verification to the so-called performance anomaly effect [8]. Interestingly, however, we found that the rate avalanche effect could be effectively ameliorated through turning on the RTS/CTS mechanism.

The discovery has motivated us to thoroughly investigate the effect of RTS/CTS exchange on the performance of multi-rate WLANs. The investigation however is non-trivial. The complexity of time-varying channel characteristics, node mobility, various traffic patterns, node interactions, and their effects on the behavior of ARF algorithm make it difficult to build a complete analytical model to study the network performance. On the other hand, a real testbed that contains a large number of wireless nodes is expensive. Therefore, we have chosen to use the Network Simulator – ns-2 [21] to conduct the study. However, the current implementation of ns-2, e.g., the recent all-in-one version 2.29, does not satisfy our requirements – while it has a rather complete, detailed, and accurate simulation of the 802.11 MAC protocol, the support to the PHY layer is much less substantial. For example, it uses a threshold-based frame reception model, i.e., a frame is received if the received power is higher than a threshold otherwise dropped, does not always yield trustworthy simulation results. For a simulation study where link dynamics and frame errors need to be carefully addressed, a concrete and realistic PHY implementation is particularly important. To facilitate the investigation, we have made significant changes in the implementation of 802.11 MAC and PHY. Besides the support to the multi-rate and ARF algorithm, the implementation also includes two important features: (1) it has a realistic channel propagation model that is suitable for mobile nodes deployed in urban area and (2) it uses a FER (Frame Error Rate)-based reception model that reflects the error performance of 802.11a PHY modulation techniques and coding schemes.

The contributions of this work are summarized as following: (i) we investigate and report the rate avalanche effect that could significantly degrade the performance of heavily loaded multi-rate 802.11 wireless networks; the reason behind this effect is analyzed and discussed; (ii) we find that the rate avalanche effect could be effectively suppressed by turning on the RTS/CTS mechanism, which suggests a new important factor to consider on the using of the RTS/CTS in multi-rate networks and (iii) through extensive simulations using realistic channel propagation and reception models, we thoroughly studied the impact of RTS/CTS on the performance of multi-rate 802.11 networks; our investigation reveals that, if dynamic RTS/CTS exchange is to be employed, simply using a pre-configured RTS threshold will only yield sub-optimal performance; this is because the optimal RTS threshold depends on several factors, such as, the number of competing nodes, the geographic distribution of nodes, and node mobility, etc. all these factors can vary over time in real networks.

The rest of this paper is organized as follows. Section 2 presents the related work. We describe the detailed simulation modeling in Section 3. Performance analysis is presented and discussed in Section 4. Finally, Section 5 concludes the paper.

2. Related work

An ideal SINR-based rate adaptation scheme would assume the following being achieved simultaneously: (i) accurately modeling the relation between SINR and the optimal rate for throughput; and (ii) senders being able to precisely estimate the SINR that will be observed by receivers at the moment of transmission. With (i) and (ii) achieved, an optimal rate adaptation algorithm could be implemented simply by the means of a lookup table. Unfortunately, in reality, neither (i) nor (ii) is easy to achieve. A complete model for the relation between SINR and the optimal rate for throughput not only includes a radio channel model describing the relation among FER, SINR, modulation and coding schemes, but also a MAC model capturing the interactions of multiple contending stations. On the other hand, estimating time-varying channel conditions is a very challenging task, especially when dynamic interferences are considered [27]. Despite these difficulties, the optimality of such SINR-based approach has attracted significant research efforts, such as RBAR [10], OAR [29], RAF [5], and others [18,24]. Assumptions or approximations are often used in these schemes to simplify the problem. For example, some approaches assume symmetric wireless links, or adapt rates based on SNR (thus ignore interferences) or received signal strength (RSS) measurement (with both noise and interferences ignored). In RBAR [10], a widely known rate adaptation protocol, receivers

estimate the channel conditions based on the RTS frame from senders, make the rate decision and inform the later through a customized CTS frame (thus incompatible to the current 802.11 specifications). It also uses a simplified, a priori channel model to calculate SINR thresholds. These SINR-based approaches have not been applied in practice so far.

An alternative to SINR-based approaches is to estimate link conditions through maintaining statistics about the transmitted data like FER [4], the achieved throughput [3], consecutive transmission successes/losses [11,19,12,9], and short-term lose ratio [30]. ARF [11], the first documented rate adaptation algorithm which was originally designed for Lucent Technologies' WaveLAN-II WLAN devices, belongs to this category. ARF incrementally increases or decreases the rate by keeping the track of acknowledged/unacknowledged transmissions as well as a timing function. A sender will switch its transmission rate to the next lower level if it experiences two consecutive failed (unacknowledged) transmissions, a timer will also be started. When either the timer expires or it sees 10 consecutive successful (acknowledged) transmissions, the sender will raise its rate to the next higher level and the timer is cancelled. However, if the first transmission at the higher rate fails, the rate will be lowered down again instantly and the timer is restarted. Some deficiencies of ARF have been revealed by previous research efforts: it is unable to adapt effectively to fast-changing channel conditions [10]. On the other hand, if the channel conditions do not change at all, or change very slowly, ARF will try to use a higher rate every 10 consecutive transmission successes or after a timer expiration, which results in increased retransmission attempts and thus harms the throughput [19]. However, due to its simplicity, ARF is still the most widely implemented rate adaption scheme in the 802.11 market [12].

In infrastructure-based WLANs, since the hidden node problem is often much less severe than in ad hoc networks, it is a common belief that it might not worth to employ the RTS/CTS exchange due to its transmission overhead. This has led to the fact that the RTS/CTS exchange has been effectively disabled (by setting a large RTS threshold value, e.g., 3000 octets) in most today's 802.11 WLANs.

Besides the transmission overhead and its effectiveness to solve hidden node problem, other possible effects of the RTS/CTS exchange have been rarely studied. One exception is [2], where Bianchi in this well-known work pointed out that in a heavily contending WLAN environment, the RTS/CTS exchange might help to reduce collisions and therefore enhance network performance even when no hidden node presents. However, Bianchi's analytical model assumed ideal channel conditions (i.e., error-free links) and only single-rate transmissions were considered.

Data frames could be lost in the air due to two reasons: (1) link errors, mainly caused by the low SINR values or (2) transmission collisions. Interestingly, the 802.11 MAC protocol and ARF have totally different assumptions on frame losses. DCF assume all frame losses are due to collisions therefore CW is doubled to reduce contention upon a frame loss; while ARF ascribes each single frame loss to a link error therefore it lowers the data transmission rate whenever consecutive frame losses happen. Obviously, either of the two explanations on frame loss may lead to network performance degradation: a too large CW that does not match the contention level may unnecessarily suppress frame transmissions [2], while a too low transmission rate will certainly waste precious channel resources. Ironically, these two opposite explanations co-act in most today's 802.11-based terminals.

Several approaches have been proposed for loss differentiation [1,22,23]. In [22], Pang et al. discussed the using of RTS/CTS exchange for loss differentiation, however, without details and results. Besides that, Kim et al. in their work [12] proposed to use RTS probing to distinguish frame losses. While these studies both mentioned the benefit and transmission overhead of using RTS/CTS exchange, none of them gave a detailed study on the effects of RTS/CTS in multi-rate 802.11 networks under complex network conditions.

3. Simulation modeling

To facilitate our investigation, we made significant changes on the implementation of 802.11 MAC and PHY in ns-2. Besides the support to the multi-rate and ARF algorithm, the implementation also includes two important features: (1) it has a realistic channel propagation model that is suitable for mobile nodes deployed in urban area and (2) it uses a FER-based reception model that reflects the error performance of 802.11a PHY modulation techniques and coding schemes.

We present the simulation modeling in details from the following four aspects: the multi-rate network model and parameters, the channel propagation model, the frame reception model, and the simulation scenarios and traffic model.

3.1. Multi-rate network model and parameters

Sharing the same protocol at the MAC layer, currently deployed 802.11 wireless devices follow different standards at the PHY layer, for example, 802.11a, 802.11b, 802.11g, 802.11n, etc. All these standards support multi-rate transmissions. For the simulation purpose, however, we have chosen to use 802.11a at the PHY layer, which supports up to eight transmission rates (6–54 Mbps), yet it is less complex than 802.11g which must support the four transmission rates of 802.11b to keep the compatibility.

IEEE 802.11a employs OFDM (Orthogonal Frequency Division Multiplexing) modulation technology in PHY layer. The basic principle of OFDM is to divide a high-speed binary signal to be transmitted over a number of low data rate subcarriers. There are a total of 52 subcarriers, of which 48 subcarriers carry actual data and four subcarriers are pilots that facilitate phase tracking for coherent demodulation. Each low data rate bit-stream is used to modulate a separate subcarrier within

Table 1
Eight PHY modes of IEEE 802.11a

Mode	Modulation	Code rate	Data rate (Mbps)
1	BPSK	1/2	6
2	BPSK	3/4	9
3	QPSK	1/2	12
4	QPSK	3/4	18
5	16-QAM	1/2	24
6	16-QAM	3/4	36
7	64-QAM	2/3	48
8	64-QAM	3/4	54

BPSK: Binary Phase-Shift Keying; QPSK: Quadrature Phase-Shift Keying; QAM: Quadrature Amplitude Modulation.

Table 2
IEEE 802.11a MAC/PHY channel parameters used in simulations

Parameters	Values
Preamble length	16 μ s
PLCP header length	4 μ s
MAC header size	28 Bytes
Slot time	9 μ s
Short Inter Frame Space (SIFS)	16 μ s
DCF Inter Frame Space (DIFS)	34 μ s
Minimum contention window size (CW _{min})	31
Maximum contention window size (CW _{max})	1023
Clear Channel Assessment (CCA) Time	3 μ s
RxTxTurnaround time	1 μ s
RTS threshold	0 or 3000 octets
Fragmentation threshold	2100 octets
LongRetryLimit	7
ShortRetryLimit	7

a 20 MHz channel in the 5 GHz frequency band. As shown in Table 1, by using different modulation schemes, such as BPSK, QPSK, 16-QAM, 64-QAM, and different coding rates, 802.11a supports eight transmission rates. The support of 6, 12, and 24 Mbps data rates (called Basic Data Rates) are mandatory. To guide against data loss, FEC (Forward Error Correction) coding/decoding is added in 802.11a PHY, which is performed by bit interleaving and convolutional coding.

Table 2 summarizes some major MAC/PHY parameters that we used in simulations. Just a comment on RTS threshold: when it is set as 0, RTS/CTS exchange will be used for all data frames. On the contrary, to disable RTS/CTS exchange sequence, we simply set the RTS threshold to 3000 octets, which is larger than any legal MAC data Frame.¹

3.2. Channel propagation model

In ns-2, Friis free-space model is used to simulate short distance propagations and two-ray-ground model is used for long distances. Although widely used, these two models are inherently limited – they assume ideal propagation conditions without considering multipath fading effects,² which often dominates the propagation effects in mobile networks. An enhanced model – the shadowing model was introduced in ns-2 to characterize the multipath fading effects. However, as pointed out in [6], using an added Gaussian random variable to simulate the fluctuations of the channel lacks experimental validation. On the other hand, extensive measurement campaign has shown that the signal deviation can be better modeled using a Rayleigh or Ricean distribution. Ricean distribution is used when there is a dominant stationary signal component present, such as a line-of-sight propagation path; while Rayleigh distribution is more suitable if such a component does not present [28].

To simulate a mobile network that is deployed in an urban area, we developed a new implementation of the 802.11 PHY layer in ns-2, which combines the log-distance path loss model [28] and the Ricean propagation model [25].

With log-distance path loss model, and assuming the transmitter–receiver separation distance is d , the received power (in dBm) of a frame is computed as

¹ As defined in IEEE 802.11, the length of an MSDU (MAC Service Data Unit) must be less than or equal to 2304 octets. Therefore, the maximum length of a MAC data frame, or MPDU (MAC Protocol Data Unit) will be 2332 (2304 plus the MAC header, which is 28 octets in length).

² Multipath fading, or called small-scale fading, is used to describe the rapid fluctuations of the amplitudes, phases, or multipath delays of a radio signal over a short period of time or travel distance, so that large-scale path loss effects may be ignored [28].

$$Pr_d[\text{dBm}] = Pr_{d_0}[\text{dBm}] + 10 \cdot n \cdot \log\left(\frac{d}{d_0}\right), \quad (1)$$

where n is the path loss exponent which indicates the rate at which the path loss increases with distance, d_0 is the close-in reference distance which is determined from measurements close to the transmitter, and Pr_{d_0} (in watt) can be calculated using the Friis free-space model:

$$Pr_{d_0}[\text{watt}] = \frac{P_t G_t G_r \lambda^2}{(4\pi)^2 d_0^2 L}, \quad (2)$$

where P_t is the transmitted power, G_t and G_r are the antenna gains for the transmitter and receiver, respectively, L is the system loss factor not related to propagation ($L \leq 1$), and λ is the radio wavelength in meters. In our simulations, the path loss exponent n is set as 3 to simulate an urban area environment, and d_0 is set as 1 m.

With Ricean propagation model, the envelop of the received signal r has the following probability density function (pdf):

$$p(r) = \begin{cases} \frac{r}{\sigma^2} e^{-\frac{r^2+A^2}{2\sigma^2}} I_0\left(\frac{Ar}{\sigma^2}\right), & \text{for } (A \leq 0, r \leq 0), \\ 0, & \text{for } (r < 0), \end{cases} \quad (3)$$

where the parameter A denotes the peak amplitude of the dominant signal component and $I_0(\cdot)$ is the modified Bessel function of the first kind and zero-order. The Ricean distribution is often described in terms of a parameter K (called Ricean factor) which is defined as the ratio between the deterministic signal power and the variance of the multipath. It is given by $K = A^2/(2\sigma^2)$ or, in terms of dB, K (dB) = $10 \log(A^2/2\sigma^2)$. As $A \rightarrow 0$, $K \rightarrow -\infty$ dB, the Ricean distribution degenerates to a Rayleigh distribution [28].

Punnoose et al. in [25] proposed an efficient implementation of Ricean propagation model based on a simple table lookup, which has been integrated into the implementation of our combined model. With the Pr_d calculated in (1), the final received power after Ricean fading effect will be

$$Pr_d^*[\text{dBm}] = Pr_d[\text{dBm}] + 10 \log(F_{\text{Ricean}}), \quad (4)$$

where F_{Ricean} is the envelope multiplicative factor resulted by the Ricean propagation model. In our simulations, the transmitted power and the Ricean factor K were set as 15 dBm and 6 dB, respectively. Node mobility is simulated using the widely adopted Random Waypoint Mobility Model (RWPM) [16].

3.3. Frame reception model

The frame reception model used in ns-2 is a simple threshold-based model: if the received power of the frame falls below the carrier sense threshold – $CS_{\text{Threshold}}$, the frame is discarded as noise to simulate not being sensed. If the received power is above $CS_{\text{Threshold}}$, however lower than the receive threshold – $Rx_{\text{Threshold}}$, the frame is marked as error and discarded. Otherwise, if the received power is above $Rx_{\text{Threshold}}$, the frame is received without errors. This reception model is obviously over-simplified. In reality, the error performance of 802.11 devices is much more complicated: besides platform-dependent factors (i.e., variations in transmitter power, antenna gain, reception sensitivity, and frequency control, etc.) and external interferences, it also highly depends on the underlying PHY modulation modes and coding schemes.

Towards a more realistic reception model, a FER-based model was set up in our simulations. The procedure of the frame receiving is sketched in Fig. 1. If the received power of a frame is above $Rx_{\text{Threshold}}$ and not discarded because of the capture effect, instead of being delivered to MAC layer instantly, it has to go through the following steps:

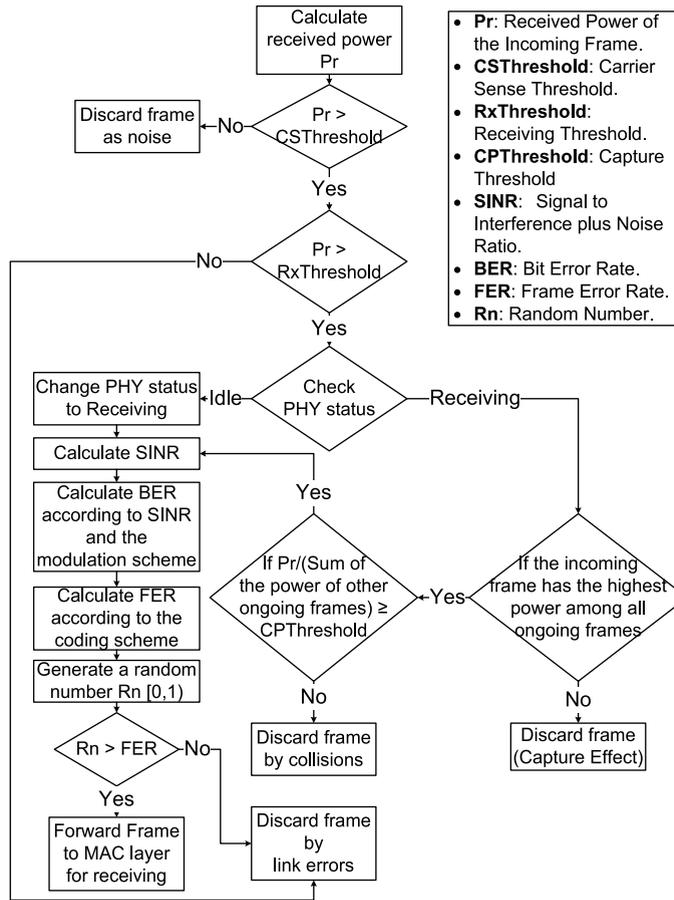
- (1) Calculate the SINR for the incoming frame using the following formula:

$$\text{SINR} = \frac{Pr_i}{\text{NoiseFloor} + \sum_{j=1, j \neq i}^n Pr_j}, \quad (5)$$

where Pr_i is the received power for the frame that we care, and $\sum_{j=1, j \neq i}^n Pr_j$ is the sum of the received power of all other ongoing frames, taken as interferences to the incoming frame.

- (2) Calculate BER (Bit Error Rate) based on the SINR and the transmission mode (see Table 1) with which the frame was sent. An AWGN (Additive White Gaussian Noise) channel noise model is assumed in the calculation.
- (3) Calculate FER using the upper bound probability of error that is given in [26] under the assumption of binary convolutional coding and hard-decision Viterbi decoding.
- (4) Generate a random number uniformly in the range of [0, 1). If the number is larger than FER, the frame is correctly received (to be delivered to the MAC layer); otherwise, the frame is marked as error and then discarded.

Taking a similar approach as in [18], we calculate the BER and FER based on the modulation and coding schemes used in 802.11a OFDM PHY, with an AWGN channel assumed.



- **Pr**: Received Power of the Incoming Frame.
- **CSThreshold**: Carrier Sense Threshold.
- **RxThreshold**: Receiving Threshold.
- **CPTreshold**: Capture Threshold
- **SINR**: Signal to Interference plus Noise Ratio.
- **BER**: Bit Error Rate.
- **FER**: Frame Error Rate.
- **Rn**: Random Number.

Fig. 1. The FER-based frame reception procedure.

3.3.1. The calculation of BER

First, let us calculate the BER for the modes listed in Table 1. For modes that use BPSK and QPSK modulation schemes, i.e., mode 1–4, the BER can be calculated using the following formula:

$$P_b^{PSK} = Q\left(\sqrt{\frac{2E_b}{N_0}}\right), \tag{6}$$

where E_b is the energy per information bit in the received signal and $N_0/2$ is the two-sided spectral density of the noise. The Q-function is defined as

$$Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-t^2/2} dt, \tag{7}$$

and E_b/N_0 can be calculated using the following formula:

$$E_b/N_0 = SINR \times \frac{SignalSpread \text{ (Hz)}}{DataRate \text{ (bps)}}, \tag{8}$$

where SINR is calculated using (5), and Signal Spread is the bandwidth of the channel (i.e., 20 MHz for 802.11a). For modes that use 16-QAM and 64-QAM modulation schemes, i.e., mode 5–8 in Table 1, with a Gray-coded assignment of bits to symbols, the BER can be approximated by

$$P_b^{QAM}(M) = 1 - \left[1 - \frac{4}{\log M} \left(1 - \frac{1}{\sqrt{M}} \right) Q\left(\sqrt{\frac{3 \log M E_b}{M - 1 N_0}} \right) \right], \tag{9}$$

where M is equal to 16 for 16-QAM, and 64 for 64-QAM. The calculation of E_b/N_0 follows the same formula as in (8).

3.3.2. The calculation of FER

The FER is calculated using an upper bound given in [26], with the assumption of binary convolutional coding and hard-decision Viterbi decoding with independent errors at the channel input. For an l -octet long chunk of data to be transmitted using PHY mode m , the error probability is bounded as

$$P_e^m(l) \leq 1 - (1 - P_u^m)^{8l}, \quad (10)$$

where the union bound P_u^m of the first-event error probability is given by

$$P_u^m = \sum_{d=d_{\text{free}}}^{\infty} a_d \cdot P_d, \quad (11)$$

where d_{free} is the free distance of the convolutional code selected in PHY mode m , a_d is the total number of error events of weight d , and P_d is the probability that an incorrect path at distance d from the correct path is chosen by the Viterbi decoder. When hard-decision decoding is applied, P_d is given by

$$P_d = \begin{cases} \sum_{k=(d+1)/2}^d \binom{d}{k} \cdot \rho^k \cdot (1-\rho)^{d-k}, & \text{if } d \text{ is odd,} \\ \frac{1}{2} \cdot \binom{d/2}{d} \cdot \rho^{d/2} \cdot (1-\rho)^{d/2} + \sum_{k=d/2+1}^d \binom{d}{k} \cdot \rho^k \cdot (1-\rho)^{d-k}, & \text{if } d \text{ is even,} \end{cases} \quad (12)$$

where ρ is the BER for the transmission mode given by (6) and (9). It is worthwhile noting that the PLCP (Physical Layer Convergence Procedure) header of an 802.11a frame, except the SERVICE field [14], is always transmitted in mode 1, while the rest of the frame might be transmitted in other modes. We calculated the error rate for each of these two parts then integrated them together to get the BER for each frame.

3.4. Simulation scenarios and traffic model

Our simulations focus on infrastructure-based 802.11a WLANs. Considering the effects of RTS/CTS exchange on the hidden node problem is well-known, in our studies, we limit nodes in a relative small area compared to their carrier sensing range,³ so that no hidden node presents. By doing this, we could focus our study on the effects of RTS/CTS exchanges on the rate adaptation itself. In different scenarios, we have up to 60 nodes randomly deployed in a square area of 60 m × 60 m, 80 m × 80 m, or 100 m × 100 m. Node mobility is simulated using the Random Waypoint Mobility Model (RWPM), with different maximal moving speeds, from 0 (i.e., static) to 10 m/s, in different scenarios.

In the simulation, traffic flows are symmetric in the sense that all of them share the same traffic load in each scenario. Traffic flows are between non-AP nodes and the AP with non-AP nodes being the senders. Our investigation mainly focuses on heavily loaded networks because of the following reasons: (i) the performance of a network is often of less interest and importance if the load is below its capacity; (ii) on the other hand, the transmission overhead of RTS/CTS exchange is not important any more when the network load is light and (iii) with the ever-growing multimedia applications, such as VoIP, video-conferencing, IPTV, and mobile gaming, etc. WLANs could often be heavily loaded or flash crowded [17].

Both UDP and TCP applications are studied. For UDP applications, the aggregated traffic load in the network could simply be expressed as

$$E(n, l, \tau) = n \times l \times (1/\tau),$$

where n represents the number of nodes, l represents the average data packet size,⁴ and τ represents the average packet interval. We vary the traffic load and patterns through changing the n , l , and τ , respectively. For the packet size, we use the range of [64, 2048]. The MTU (Maximum Transmission Unit) is set to 2304 Bytes, which is the maximum frame body length defined in IEEE 802.11 standard [13]. So in this case, no IP fragmentation is done.

Besides the above traffic model, for the study of UDP applications, we also use a trace of real H.263 video stream captured in an office environment [7]. This trace produces a variable bit rate (VBR) traffic with a target bit rate of 256 kbits/s. The packets contained have the average length of 6199 Bytes, with the min and max of 864 Bytes and 8514 Bytes, respectively. Different from the above simple UDP traffic model, we have conservatively⁵ chosen a smaller value – 1500 Bytes – for the MTU, to cover the situations where path MTU discovery [20] is used. Most packets will be fragmented in this case.

TCP applications, on the other hand, are self-adaptive, for which we only vary the average packet size. TCP NewReno with selected ACKs is used in our simulations.

³ With our propagation model and the setting of $CSThreshold$ (−96 dB), the carrier sensing range is about 135 m when all nodes are static.

⁴ The data packet size referred in this work includes the length of UDP and IP headers, however, not the MAC header.

⁵ As shown later, the rate avalanche effect has more significant influence on network performance for longer frames.

4. Performance analysis

Various scenarios have been simulated to thoroughly study the effects of RTS/CTS exchange on the performance of multi-rate 802.11 networks. For each scenario, we compare the network performance for the case with RTS/CTS turned on and the case with RTS/CTS turned off. The purpose of the study is to answer the following questions:

- Is RTS/CTS exchange superfluous in multi-rate WLANs when no hidden node presents?
- If the answer to the first question is no, how does the RTS/CTS exchange influence the network performance under various scenarios and network conditions?

Different from most work on rate adaptation, we chose to use network-wide aggregate throughput, instead of per-flow throughput, as the major performance metric. Two reasons are behind: (1) the performance anomaly of multi-rate networks (see Section 2) – a per-flow optimal strategy might lead to the performance degradation of the whole network; and (2) the network revenue is better represented by the network-wide aggregate throughput than the throughput of some individual flows. With that said, we also paid attention to individual flows through examining the fairness issue in Section 4.6.

As previously stated, ARF protocol is used to adapt transmission rates in the simulation. However, we believe the insights provided here also apply to other statistics-based rate adaptation schemes.

4.1. The rate avalanche effect and the impact of packet size

First, let us use one example scenario to demonstrate the effect that we call rate avalanche. In this scenario, 40 802.11a wireless nodes are randomly deployed in a square area of 80 m × 80 m and one AP is installed in the center of the area (i.e., (40,40)). There is one UDP-based traffic flow from each node to the AP. Each flow has the packet arrival rate of 200 packets/s (or a packet interval of 5 ms). All the flows start at 0 s and stop at 30 s. Nodes are kept static during the whole simulation procedure. We first turn RTS/CTS on and repeat the experiment with different packet sizes, from 64 Bytes to 2048 Bytes with the step of 64 Bytes. Using the exactly same settings, the same group of experiments are repeated with RTS/CTS turned off.

Fig. 2 compares the aggregate throughput for the cases with RTS/CTS turned on and the cases with RTS/CTS turned off (referred as RTS/CTS-on and RTS/CTS-off, respectively, in the rest of the paper). As we can see from the figure, between 64 Bytes and 640 Bytes, RTS/CTS-on delivers slightly lower throughput than RTS/CTS-off. However, starting from 640 Bytes, the performance of RTS/CTS-off are much lower than that of RTS/CTS-on.

With no hidden node presenting, it was expected the RTS/CTS-off would always deliver higher throughput than RTS/CTS-on due to the transmission overhead of RTS/CTS exchanges. So what causes the performance degradation of RTS/CTS-off? The question can be answered by Figs. 3 and 4, which compare the number of collisions and the cumulative distribution of transmission rates used for RTS/CTS-on and RTS/CTS-off, respectively, (for the experiment with packet size of 1024 Bytes): (1) Fig. 3 demonstrates that RTS/CTS-on leads to fewer collisions. This is because RTS frame is much shorter compared to most data frames, thus the collision probability of RTS frames is much lower. And once a RTS frame is successfully transmitted, the channel access is then reserved for the CTS and data frames that follow. Less collisions lead to less retransmissions which save the channel resources. (2) More importantly, Fig. 4, with rates 1–8 corresponding to 6–54 Mbps, respectively, clearly

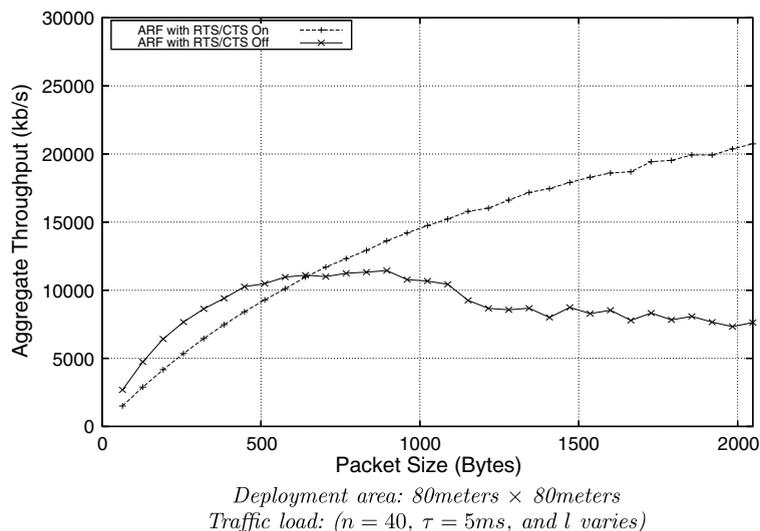


Fig. 2. Aggregate throughput: RTS/CTS-on vs. RTS/CTS-off. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ = 5 ms, and l varies).

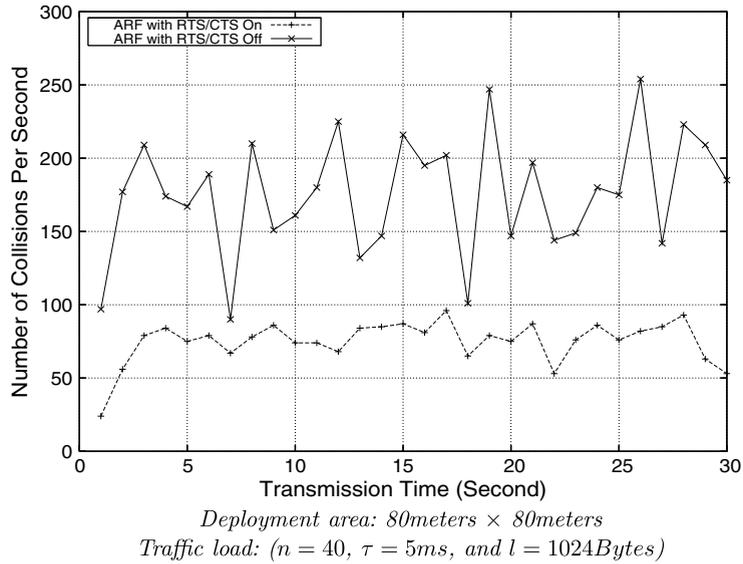


Fig. 3. Number of collisions per second. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ = 5 ms, and l = 1024 Bytes).

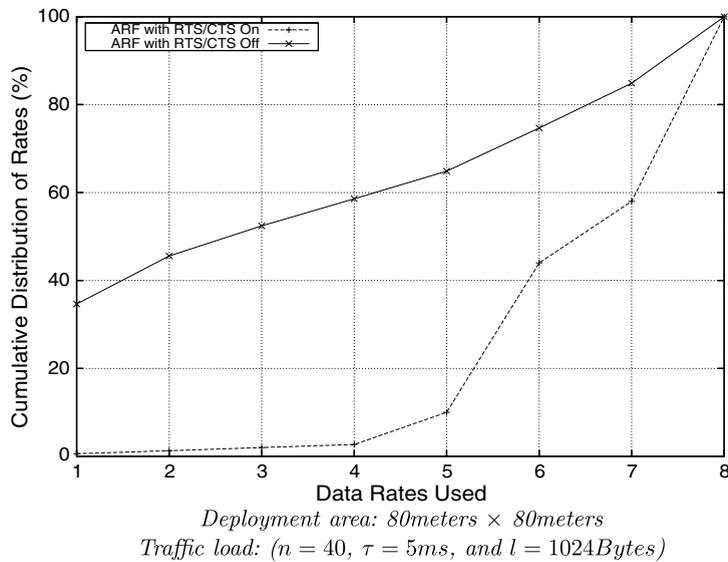


Fig. 4. Cumulative distributions of data rates used. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ = 5 ms, and l = 1024 Bytes).

shows that RTS/CTS-off leads to a much higher percentage of lower rate transmissions than RTS/CTS-on. ARF behaves differently because the only feedback that ARF used to adapt transmission rates is the statistics of acknowledged/unacknowledged data transmissions. When RTS/CTS is off, every collision will cause an unacknowledged transmission, which will lead to rate dropping; on the other hand, when RTS/CTS is on, most collisions will be RTS frame collisions, which only lead to retransmissions of RTS frames, instead of unacknowledged data transmissions. Figs. 5 and 6 give more details about rates used over time for RTS/CTS-on and RTS/CTS-off, respectively, while Fig. 7 shows the cumulative distribution of the SINR measured at the receiver for all the frames received with or without errors, indicating that SINR should not be explained as the reason of the discrepancy in rates.

It is worthwhile noting that, as the RTS/CTS transmission overhead is always incurred for all the cases when RTS/CTS is turned on, so does the rate avalanche for RTS/CTS-off; however, they are differently influenced by the (average) size of the data packets. While the effect of RTS/CTS transmission overhead becomes less severe when the packet size increases, the rate avalanche effect goes to the opposite direction – it is amplified with relatively larger data packets. This explains the crossing of the two curves in Fig. 2. The packet size corresponding to the **cross-point** could be used as the optimal RTS Threshold.

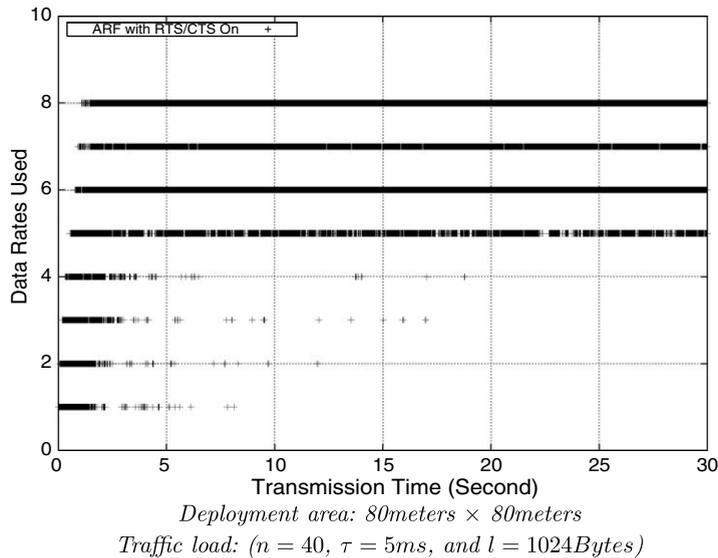


Fig. 5. Data rates used for RTS/CTS-on. Deployment area: 80 m × 80 m; traffic load: ($n = 40$, $\tau = 5$ ms, and $l = 1024$ Bytes).

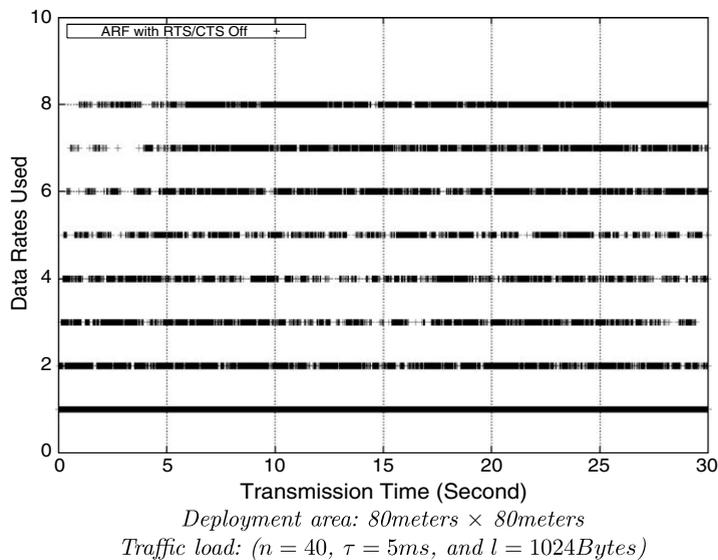


Fig. 6. Data rates used for RTS/CTS-off. Deployment area: 80 m × 80 m; traffic load: ($n = 40$, $\tau = 5$ ms, and $l = 1024$ Bytes).

Another important observation from Fig. 2 is while the performance gain of turning RTS/CTS on when packets are larger than the cross-point value is very significant, the performance gain of turning RTS/CTS off for packets that are smaller than the cross-point value is much less substantial.

4.2. The impact of node numbers

To study the impact of node numbers, in this group of experiments, we vary the number of nodes from 20 to 60. As in previous subsection, nodes are randomly deployed in a square area of 80 m × 80 m with one AP placed in the center. UDP-based applications are used for all experiments with a fixed packet interval of 5 ms. As shown in Fig. 8, the cross-points highly depend on the number of nodes. When the number of nodes gets larger, the cross points (or the optimal RTS thresholds) are shifting toward smaller packet size, which means the chance that RTS/CTS-on outperforms RTS/CTS-off is getting larger. This could be easily understood: higher number of nodes yield higher level of contentions and more collisions, which lead to more retransmissions and deeper degree of rate avalanche.

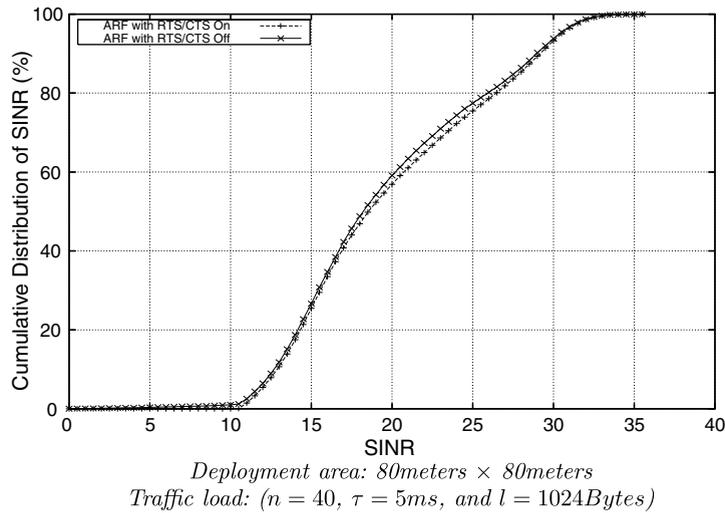


Fig. 7. Cumulative distribution of the SINR. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ = 5 ms, and l = 1024 Bytes).

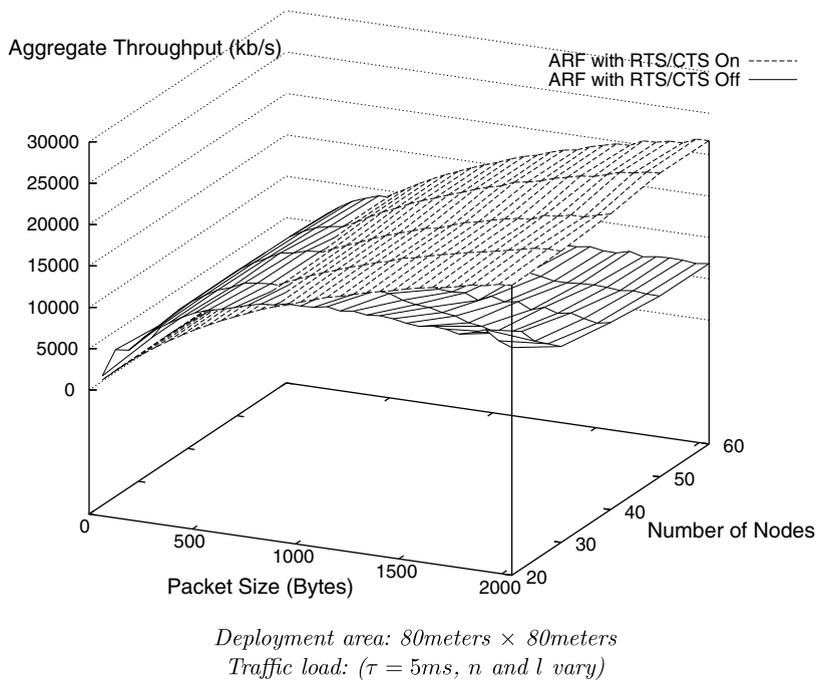


Fig. 8. The impact of node numbers. Deployment area: 80 m × 80 m; traffic load: (τ = 5 ms, n and l vary).

4.3. The impact of packet arrival rate

Next, we vary the packet arrival rate to see how the traffic load influences the performance of RTS/CTS-on and RTS/CTS-off. To focus on the area around cross-points, we only present the results for the packet size varying from 64 Bytes to 1024 Bytes. The result is quite interesting as shown in Fig. 9. As the results for τ = 5 ms and τ = 10 ms are similar to each other, the cases for larger τ values demonstrate a different pattern: their cross-points are shifting toward larger packet sizes quickly as the τ increases (i.e., packet arrival rate decreases). More interestingly, the performance gain of RTS/CTS-off over RTS/CTS-on (before the cross-points) is almost neglectable. The reason behind this is that the network starts to be under-loaded when τ is larger than 15 ms, especially for the cases with small packet sizes. And as τ increases, the under-loaded area is spreading to larger packet sizes. Compared to over-loaded situations, both the performance gain of RTS/CTS-off (before the cross-points) and RTS/CTS-off (after the cross-points) for an under-loaded network are much less significant.

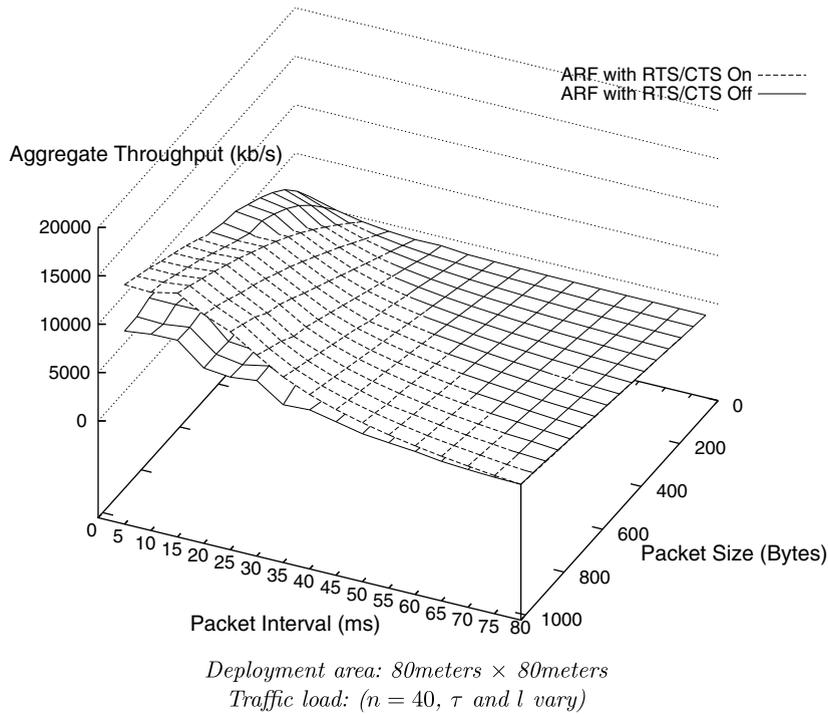


Fig. 9. The impact of packet arrival rate. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ and l vary).

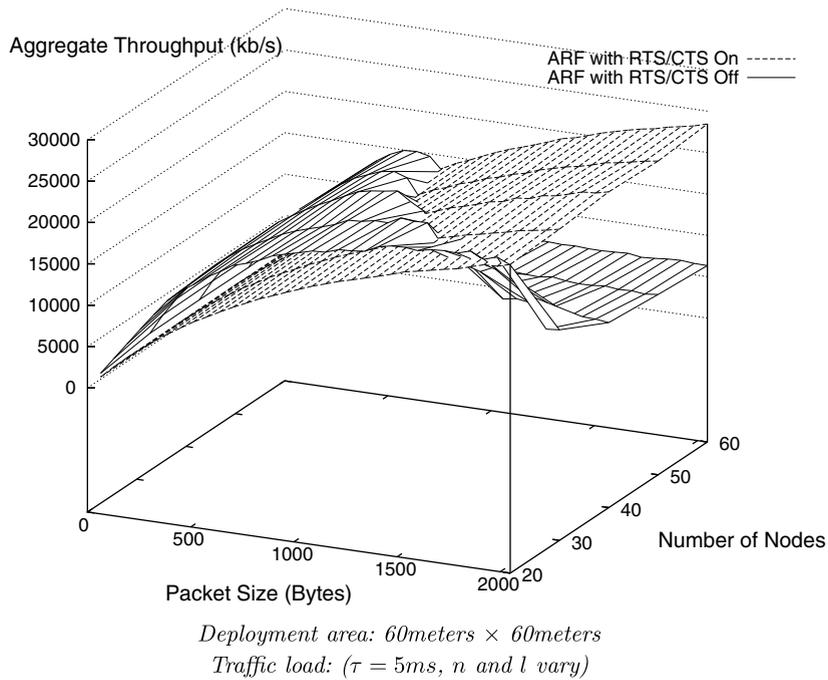


Fig. 10. The impact of node geographical distributions (A). Deployment area: 60 m × 60 m; traffic load: (τ = 5 ms, n and l vary).

4.4. The impact of node geographical distributions

In all the scenarios presented previously, nodes are randomly deployed in a square area of 80 m × 80 m. To study how geographical distribution could influence the performance of RTS/CTS-on and RTS/CTS-off, we varied the size of the node

deployment area. Figs. 10 and 11 show the results for the cases of $60\text{ m} \times 60\text{ m}$ and $100\text{ m} \times 100\text{ m}$, respectively. Comparing these two figures with Fig. 8, we can see that as the deployment area is larger (i.e., nodes are more spreading out), the cross-points (i.e., the optimal thresholds) are shifting toward smaller packet sizes. This could be explained as follow. When nodes are deployed in a wider area, the chances that they have to use lower rates for transmission are getting higher. The side-effect is that the relative overhead of transmitting RTS/CTS exchanges is getting lower since the difference between the transmission rates for data frames and RTS/CTS exchanges are smaller.

Beside that, we also notice that when the deployment area is larger, the performance gain of RTS/CTS-off over RTS/CTS-on before the cross-points in the curves, diminishes quickly.

4.5. The impact of node mobility

To investigate how node mobility might influence the performance of RTS/CTS-on and RTS/CTS-off, we compare the results for experiments that use three different Maximal Velocity values, namely, 2, 5 and 10 m/s. The Maximal Velocity is a parameter defined in RWPM mobility model to bound the random moving speed of a mobile node. In this group of experiments, 40 nodes are first randomly deployed in the area of $80\text{ m} \times 80\text{ m}$, then move individually following the RWPM model. Results are shown in Fig. 12. As we can see from the figure, a higher moving speed causes a slight drop in the performance of both RTS/CTS-on and RTS/CTS-off. The cross-point also shifts toward smaller packet sizes when the moving speed increases. This is because, overall, a faster moving speed of the sender, the receiver and other surrounding nodes will also yield faster fluctuations of the received signal strength at the receiver due to the Doppler shifts and multipath propagations. Reacting to a faster changing channel is surely more challenging to the ARF scheme.

4.6. Mixed RTS/CTS-ons and offs

In previous experiments, the action of the nodes is uniform – either all of them have RTS/CTS turned on or all of them have RTS/CTS turned off. Next we investigate the cases that nodes are not uniform – a portion of the nodes have RTS/CTS-on while the rest have RTS/CTS-off. With a total of 40 nodes (moving with the maximal velocity of 5 m/s with RWPM model), we vary the number of nodes that have RTS/CTS-on from 5 to 35; correspondingly, the number of nodes that have RTS/CTS-off are from 35 to 5. For example, a combination of (15,25) means 15 of the flows have RTS/CTS turned on (referred as on-flows) while the rest have it turned off (referred as off-flows). Seven combinations, namely (5, 35), (10,30), (15,25), (20,20), (25,15), (30,10), and (35,5), have been investigated. Fig.13 compares the aggregate throughput of the seven combinations. The aggregate throughput for the cases of (40,0) (i.e., RTS/CTS-on) and (0,40) (i.e., RTS/CTS-off) are used as the references in the figure. Without much surprise, the results of these mixed cases are bounded by the RTS/CTS-on and RTS/CTS-off. With more off-flows, RTS/CTS-on degenerates smoothly to RTS/CTS-off. It is worthwhile noting that the cross-point value (i.e., the optimal RTS threshold) does not have noticeable changes among different combinations.

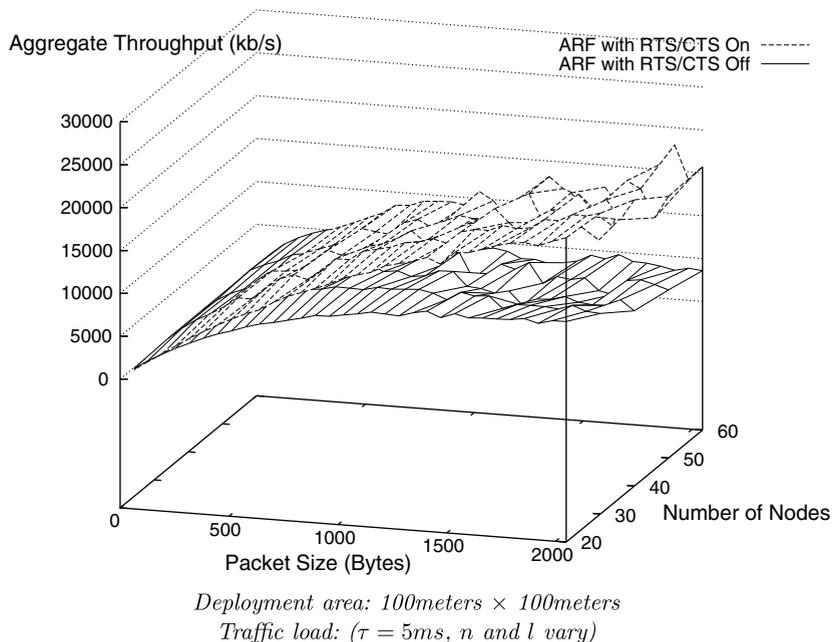


Fig. 11. The impact of node geographical distributions (B). Deployment area: $100\text{ m} \times 100\text{ m}$; traffic load: ($\tau = 5\text{ ms}$, n and l vary).

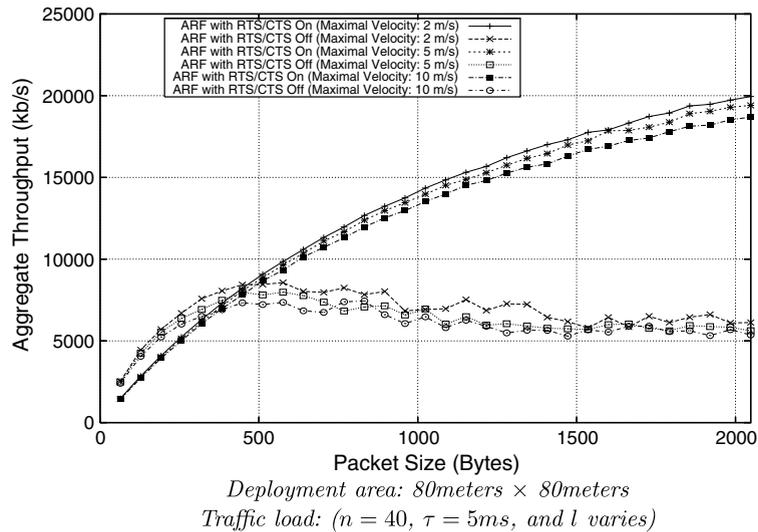


Fig. 12. The impact of node mobility. Deployment area: 80 m \times 80 m; traffic load: ($n = 40$, $\tau = 5$ ms, and l varies).

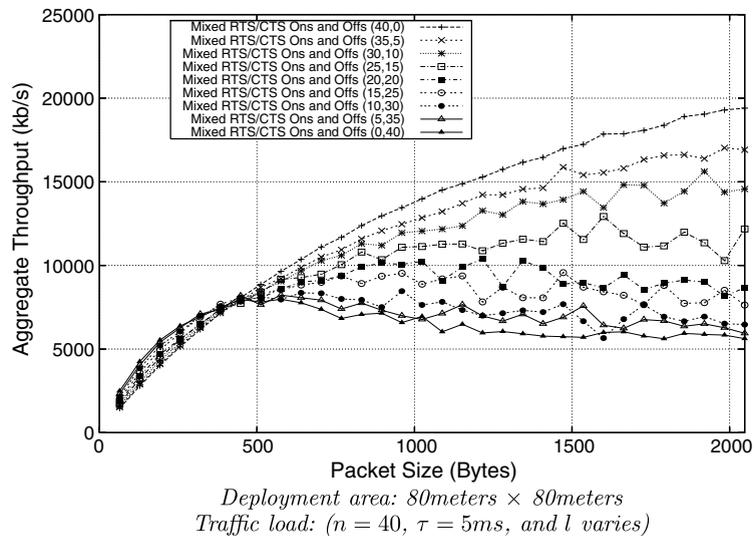


Fig. 13. Mixed ons and offs. Deployment area: 80 m \times 80 m; traffic load: ($n = 40$, $\tau = 5$ ms, and l varies).

To look into more details, we average the throughput for on-flows and off-flows, respectively. As shown in Fig. 14, in all combinations, off-flows always have higher throughput than on-flows, although the difference is very small for some combinations. This result is interesting since it implies a possibility that a selfish node, if it is “near-sight” also, might select to turn off RTS/CTS all the times in order to achieve higher throughput than other competing nodes. However, by doing this, this node not only actually achieves opposite effect by lowering down its own throughput but also degrades the performance of the whole network.

4.7. Result of the H.263 video trace

In this experiment, we study the rate avalanche effect using the real H.263 video trace captured in an office environment. 40 mobile nodes are randomly deployed in a square area of 80 m \times 80 m. Again RWPM model is used to simulate their mobility and the maximal velocity is set to be 10 m/s. Each node sends two video streams to the AP. Fig. 15 compares the aggregate throughput for RTS/CTS-on and RTS/CTS-off along the time. Obviously, the rate avalanche effect severely degrades the network performance.

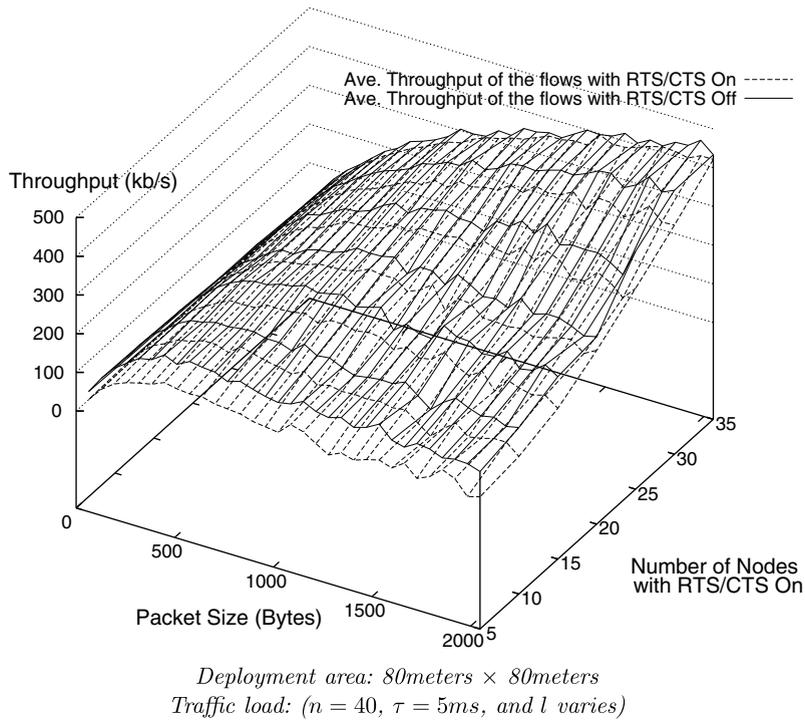


Fig. 14. Mixed ons and offs: aggregate throughput for on-flows and off-flows. Deployment area: 80 m × 80 m; traffic load: (n = 40, τ = 5 ms, and l varies).

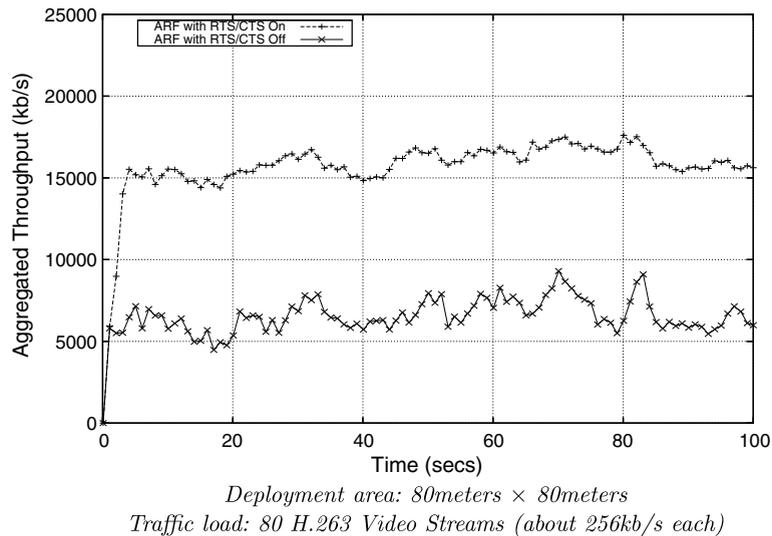


Fig. 15. Using the real H.263 video trace captured in an office environment. Deployment area: 80 m × 80 m; traffic load: 80 H.263 video streams (about 256 kb/s each).

4.8. TCP-based applications

All previously presented results are for UDP-based applications. Next we present the results for TCP-based applications. With the exactly same settings as used in the experiments reported in Fig. 2, we conducted the simulations for TCP-based applications. Particularly, TCP NewReno and selected ACKs are used. Results are shown in Fig. 16. As we can see in the figure, the performance gain of RTS/CTS-on over RTS/CTS-off after the cross-point is still significant, although not as big as for UDP-based applications. TCP-based applications are less affected by the rate avalanche effect than UDP-based applications because their self-adaptive capabilities help to reduce the contention level of the shared channel.

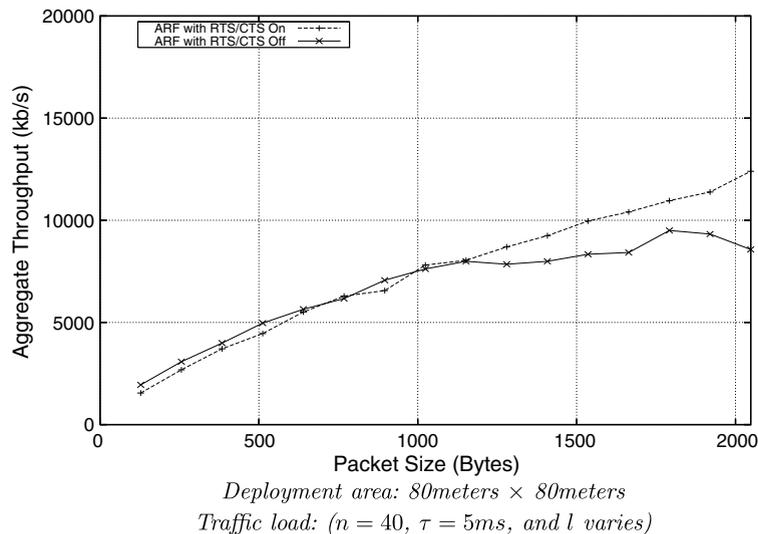


Fig. 16. TCP-based applications. Deployment area: 80 m × 80 m; traffic load: ($n = 40$, $\tau = 5$ ms, and l varies).

4.9. Summary

We summarize the findings about the impact of RTS/CTS exchange on multi-rate 802.11 WLANs as follow:

- Different than in single-rate networks, simply turning off RTS/CTS exchange in heavily loaded multi-rate 802.11 WLANs could lead to severe degradation of network performance. Particularly, the rate avalanche effect contributes significantly to the performance degradation.
- The effect of rate avalanche on TCP applications is less severe than on UDP applications, due to the TCP's self-adaptive capability.
- The impact of RTS/CTS exchange in a lightly loaded multi-rate WLANs is neglectable.
- With all factors considered, keeping RTS/CTS on will have a much higher chance to gain a better network performance than keeping RTS/CTS off.
- If dynamic RTS/CTS exchange is to be employed, simply using a pre-configured RTS threshold will only yield sub-optimal performance. This is because the optimal RTS threshold depends on several factors, such as, the number of competing nodes, the geographic distribution of nodes, and node mobility, etc. All these factors can vary over time in real networks.

5. Conclusion

With the original mission to deal with the hidden node problem, the RTS/CTS exchange has been disabled in most infrastructure-based 802.11 WLANs due to its transmission overhead. While this might be a wise strategy for single-rate networks, we argue that it could significantly hurt the network performance in multi-rate networks. Particularly, we report an important phenomenon called rate avalanche that happens in heavily loaded multi-rate 802.11 WLANs when RTS/CTS exchange is turned off. Motivated by this discovery, we conducted an extensive investigation on the effects of the RTS/CTS exchange on the performance of multi-rate 802.11 WLANs. The investigation was based on realistic channel propagation and frame reception models. We simulate various scenarios/conditions to study their impact on the network performance for RTS/CTS-on and off, respectively. Some important insights about using the RTS/CTS exchange in multi-rate 802.11 WLANs are provided.

Acknowledgements

A short version of this work was published in IPDPS–DPDNS 2008 workshop [31]. Liqiang Zhang's research was supported in part by US National Science Foundation grant CNS-0834230 and an Indiana University Faculty Research Grant. Xiaobo Zhou's research was supported in part by US National Science Foundation grant CNS-0720524. We thank the guest editors for their time, encouragements and support throughout the revision process. We are also very grateful to the anonymous reviewers for their constructive comments, which helped to improve the quality of this paper.

References

- [1] I. Aad, Quality of Service in Wireless Local Area Networks, Ph.D. Dissertation, INRIA, France, October, 2002.
- [2] G. Bianchi, Performance analysis of the IEEE 802.11 distributed coordination function, *IEEE Journal on Selected Areas in Communications* 18 (3) (2000) 535–547.
- [3] J. Bicket, Bit-rate Selection in Wireless Networks, MIT Master Thesis, 2005.
- [4] B. Braswell, J. McEachen, Modeling data rate agility in the IEEE 802.11a WLAN protocol in OPNETWORK 2001, Washington, DC, August, 2001.
- [5] C. Chen, Rate-adaptive framing for interfered wireless networks, in: Proc. of IEEE INFOCOM'07, Anchorage, Alaska, 2007.
- [6] J. Dricot, P. Doncker, High-accuracy physical layer model for wireless network simulations in ns-2, in: Proc. of International Workshop on Wireless Ad Hoc Networks (IWVAN 2004), Finland, May–June, 2004.
- [7] F. Fitzek, M. Reisslein, MPEG-4 and H.263 video traces for network performance evaluation, *IEEE Network* 15 (2001) 640–653.
- [8] M. Heusse et al., Performance anomaly of 802.11b, in: Proc. of IEEE INFOCOM 2003, San Francisco, California, April, 2003.
- [9] I. Haratcherev et al., Hybrid rate control for IEEE 802.11, in: Proc. of ACM MOBIWAC'04, Philadelphia, PA, September, 2004.
- [10] G. Holland et al., A Rate-adaptive MAC protocol for multi-hop wireless networks, in: Proc. of ACM MOBICOM'01, Rome, Italy, July, 2001.
- [11] A. Kamerman, L. Monteban, WaveLAN-II: a high-performance wireless LAN for the unlicensed band, *Bell Labs Technical Journal*, Summer, 1997, pp. 118–133.
- [12] J. Kim et al., CARA: Collision-aware rate adaptation for IEEE 802.11 WLANs, in: Proc. of IEEE INFOCOM'06, Barcelona, Spain, April, 2006.
- [13] IEEE 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11-1999, 1999.
- [14] IEEE 802.11a, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: High-speed Physical Layer in 5 GHz Band, September, 1999.
- [15] Z. Ji et al., Exploiting medium access diversity in rate adaptive Wireless LANs, in: Proc. of ACM MOBICOM'04, Philadelphia, PA, September, 2004.
- [16] D. Johnson, D. Maltz, Dynamic source routing in ad hoc wireless networks, *Mobile Computing*, Kluwer Academic Publishers, 1996, pp. 153–181.
- [17] A. Jordash et al., IQU: Practical queue-based user association management for WLANs, in: Proc. of ACM MOBICOM'06, Marina del Rey, CA, September, 2006.
- [18] D. Qiao et al., Goodput analysis and link adaptation for IEEE 802.11a Wireless LANs, *IEEE Transactions on Mobile Computing* 1 (4) (2002).
- [19] M. Lacage et al., IEEE 802.11 Rate adaptation: a practical approach, in: Proc. of ACM MSWiM'04, Venezia, Italy, October, 2004.
- [20] J. Mogul, S. Deering, Path MTU Discovery, RFC 1191, November, 1990.
- [21] The network simulator – ns-2. <<http://www.isi.edu/nsnam/ns>>.
- [22] Q. Pang et al., Design of an effective loss-distinguishable MAC protocol for 802.11 WLAN, *IEEE Communications Letters* 9 (9) (2005) 781–783.
- [23] Q. Pang et al., A rate adaptation algorithm for IEEE 802.11 WLANs based on MAC-layer loss differentiation, in: Proc. of BROADNETS 2005, Boston, MA, October, 2005.
- [24] J. Pavon, S. Choi, Link adaptation strategy for IEEE 802.11 WLAN via received signal strength measurement, in: Proc. of IEEE ICC'03, Anchorage, AK, May, 2003.
- [25] R. Punnoose et al., Efficient simulation of Ricean fading within a packet simulator, in: Proc. of IEEE VTC 2000 Fall, September, 2000.
- [26] M. Pursley, D. Taipale, Error probabilities for spread-spectrum packet radio with convolutional codes and Viterbi decoding, *IEEE Transactions on Communications* 35 (1) (1987) 1–12.
- [27] L. Qiu et al., A general model of wireless interference, in: Proc. of ACM MOBICOM'07, Montreal, Canada, September, 2007.
- [28] T.S. Rappaport, *Wireless Communications: Principles and Practice*, second ed., Prentice Hall, 2002.
- [29] B. Sadeghi et al., Opportunistic media access for multirate ad hoc networks, in: Proc. of ACM MOBICOM'02, Atlanta, Georgia, September, 2002.
- [30] S. Wong et al., Robust rate adaptation for 802.11 wireless networks, in: Proc. of MOBICOM'06, Los Angeles, CA, September, 2006.
- [31] L. Zhang, Y. Cheng, X. Zhou, Rate avalanche: the performance degradation in multi-rate 802.11 WLANs, in: Proceedings of the 22nd IEEE/ACM International Parallel & Distributed Processing Symposium (IPDPS), DPDNS'08 Workshop, Miami, FL, April, 2008.