Adaptive Packetization for Error-Prone Transmission over 802.11 WLANs with Hidden Terminals

Wei Song, Michael N. Krishnan, and Avideh Zakhor
Video and Image Processing (VIP) Lab
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, USA
Email: {wsong, mkrishna, avz}@eecs.berkeley.edu

Abstract—**Collision and fading are the two main sources of packet loss in wireless local area networks (WLANs) and as such, both are affected by the packetization at the medium access control (MAC) layer. While a larger packet is preferred to balance protocol header overhead, a shorter packet is less vulnerable to packet loss due to channel fading errors or staggered collisions in the presence of hidden terminals. Direct collisions due to backoff are not affected by packet size. Recently, Krishnan *et. al.* have developed a new technique for estimating probabilities of various components of packet loss, namely, direct and staggered collisions and fading. Motivated by this work, in this paper, we exploit ways in which packetization can be used to improve throughput performance of WLANs. We first show analytically that the effective throughput is a unimodal function of the packet size when considering both channel fading and staggered collisions. We then develop a measurement-based algorithm based on golden section search to arrive at an optimal packet size for MAC-layer transmissions. Our simulations demonstrate that packetization based on our search algorithm can greatly improve the effective throughput of sensing-limited nodes, and reduce video frame transfer delay in WLANs.**

Index Terms—**Packetization, MAC collisions, wireless LAN, video transmission**

I. INTRODUCTION AND RELATED WORK

Aiming at ubiquitous connectivity, wireless networks have been evolving rapidly during the past decade. Wireless local area networks (WLANs) based on IEEE 802.11 are widely deployed due to the low cost and ease of implementation. Originally, 802.11 WLANs were designed for best-effort data service as a wireless extension to Ethernet. Currently, there are ever-increasing demands for exploiting the existing network infrastructure to provide multimedia services, which require a large throughput and a high level of quality-of-service (QoS) assurance. With the hostile wireless channel, QoS guarantees in wireless networks are challenging because of fading, co-channel interference, and/or user mobility. The QoS provisioning in 802.11 WLANs is further limited by its contention-based medium access control (MAC).

Specifically, the 802.11 mechanism distributed coordination function (DCF) delivers highly varying service rates to the application layer. A primary reason for such variability is the collisions due to random access when multiple nodes transmit at the same time. In general, collisions can be classified into two groups: direct and staggered [1]. A direct collision occurs when more than one node simultaneously finishes backoff according to a binary exponential backoff mechanism. On the other hand, staggered collisions refer to collided transmissions from nodes that cannot sense each other's carrier, i.e., hidden terminals. The simultaneous transmissions from hidden terminals may not necessarily start exactly at the same time. Although the request-to-send (RTS) and clear-to-send (CTS) handshaking is proposed to address staggered collisions, it is rarely used in practice due to a large access delay and overhead. This delay makes the RTS/CTS mechanism particularly inappropriate for delay-sensitive real-time services such as video.

Many packetization schemes have been proposed in the literature for video services over WLANs. In H.264 video coding [2], a separate network adaptation layer (NAL) is designed to deal with application-layer packetization in order to facilitate video transmission. In [3], van der Schaar and Turaga propose cross-layer packetization and retransmission strategies to minimize video distortion, subject to a delay constraint. However, the transfer delay is approximated by incorporating an average timing overhead for the MAC layer, neglecting the delay dynamics of multiple access. In [4], optimal MAC-layer packetization is studied together with selection of the best modulation and coding rate at the physical layer. In [5], an optimal MAC-layer packet size is derived to reduce both header overhead and packet loss due to channel fading errors. However, staggered collisions with hidden terminals are considered neither in [4] nor in [5].

The main goal of this paper is to provide analytical and simulation results on MAC-layer packetization, taking into

account header overhead, packet loss due to channel fading errors as well as those due to direct and staggered collisions. In doing so, we are motivated by the recent work of Krishnan *et. al.* [1], who developed an estimation strategy for direct and staggered collision probabilities of each node in WLANs. The basic idea behind their strategy is for the access point (AP) to periodically broadcast a channel busy/idle signal to all associated nodes. Each node will then use this information together with its own local busy/idle signal in order to estimate probabilities of direct and staggered collisions locally.

The remainder of this paper is organized as follows. In Section II, we first discuss the impact of channel fading, direct collisions and staggered collisions on packetization. Then, we introduce a search algorithm to determine the optimal transmission packet size. Simulation results are presented in Section III, followed by conclusions in Section IV.

II. ADAPTIVE PACKETIZATION

A. Impact of Channel Fading Errors and Collisions

In 802.11 WLANs, the transmission packet size directly affects the achievable throughput through the contention-based MAC protocol. In [5], a throughput analysis is presented, in which only the impact of direct collisions and channel fading are considered. In this section, we also include the impact of staggered collisions in obtaining an analytical expression for throughput. Let L denote the size of application payload packetized in bytes for MAC-layer transmission. Then, the effective throughput, S_n, achievable by node n is given by

$$S_n = \frac{L \cdot P_{suc}}{\overline{D}} \qquad (1)$$

where $\overline{D}$ is the average time from that the packet enters the MAC-layer transmission queue to that it is successfully acknowledged or fails after m times of transmissions. P_{suc} is the transmission success probability of the packet, and depends on the retransmit limit m and packet loss probability, denoted by P_L, which is the probability that a packet transmission is corrupted due to channel fading, direct collisions or staggered collisions. P_{suc} can be written as

$$P_{suc} = \sum_{i=1}^{m} P_L^{i-1}(1 - P_L). \qquad (2)$$

A direct collision happens when two nodes finish backoff and start transmitting a packet at the same time. In the presence of hidden terminals, two distant nodes that cannot hear each other may experience a staggered collision even when they do not start transmission at the exactly same time. Further, channel fading errors also lead to packet loss. Hence, we have

$$P_L = 1 - (1 - P_{SC})(1 - P_{DC})(1 - P_E) \qquad (3)$$

where P_{SC} is probability that the packet to be transmitted experiences a staggered collision, P_{DC} is the probability that the packet experiences a direct collision given it does not experience a staggered collision, and P_E is the packet error

rate due to channel fading given no collisions [1]. As studied in [6], the direct collision probability, P_{DC}, only depends on the binary exponential backoff mechanism and is not affected by the packet size. In the presence of hidden terminals, staggered collisions occur when one node begins to transmit during another node's transmission [1]. Hence, the staggered collision probability, P_{SC}, increases with packet size, L, because a longer transmission duration is more susceptible to be interrupted by other hidden terminals or to interrupt other hidden nodes' transmission. Intuitively, P_{SC} is assumed to be related to the packet size L exponentially [1]:

$$P_{SC} = 1 - (1 - \tau_h)^L \qquad (4)$$

where τ_h is the transmission probability of colliding nodes that are hidden from each other at any time slot. Similarly, as shown in [5], P_E is upper-bounded by

$$P_E \leq 1 - (1 - p_b)^L \qquad (5)$$

where p_b is the bit error rate (BER) of the WLAN channel.

Incorporating both channel fading and collisions in the presence of hidden terminals, we can estimate the effective throughput of node n as follows:

$$S_n = \frac{L \cdot P_{suc}}{\displaystyle\sum_{i=1}^{m} T_i P_L^{i-1}(1 - P_L) + T_f P_L^m} \qquad (6)$$

where T_i $(i = 1, ..., m)$ is the total duration of a successful transmission after i transmission attempts, and T_f is the total duration of a finally failed transmission that reaches the retransmit limit. According to the 802.11 MAC protocol, T_i and T_f can be approximated by [6]

$$T_i = \sum_{j=1}^{N} T_{sj} + \frac{(i-1)\sum_{j=1}^{N} T_{cj}}{2} + \overline{W}, \qquad i = 1, ..., m$$

$$T_f = \frac{m \sum_{j=1}^{N} T_{cj}}{2} + \overline{W} \qquad (7)$$

where N is the total number of nodes in the WLAN, T_{sj} is the time duration of a successful transmission attempt by node j, T_{cj} is the duration that node j is devoted to one packet collision, and $\overline{W}$ is the average backoff time per packet. Here, $T_{sj} = T_{DATA} + T_{SIFS} + T_{ACK} + T_{DIFS}$ and $T_{cj} = T_{DATA} + T_{ACK\_TO} + T_{DIFS}$, where T_{DATA} is the transmission time of a `DATA` frame, T_{ACK} is the transmission time of an `ACK` frame, T_{ACK_TO} is the waiting time for an `ACK TIMEOUT`, T_{DIFS} is the duration of DCF interframe space (DIFS), and T_{SIFS} is the duration of short interframe space (SIFS). The average backoff time $\overline{W}$ is given by

$$\overline{W} = \begin{cases} \dfrac{1}{2}\displaystyle\sum_{j=0}^{i-1} 2^j W, & \text{if } i \leq m' + 1 \\[2ex] \dfrac{1}{2}\displaystyle\sum_{j=0}^{m'} 2^j W + \dfrac{(i - m' - 1)2^{m'} W}{2}, & \text{if } i \geq m' + 2 \end{cases}$$

$$(8)$$

where $W = CW_{min} + 1$ with CW_{min} being the initial back-off window size, e.g. 31 in IEEE 802.11, m is the retransmit limit, and m' is the maximum backoff stage.

It is known that packetization involves packet headers at various protocol layers. For example, operating over the physical layer and link layer, there is a commonly used protocol hierarchy for real-time services, i.e., real-time transport protocol (RTP), user datagram protocol (UDP), and Internet protocol (IP). The overall packet headers can be as long as 40 bytes, which is a non-negligible overhead. Hence, a larger packet is preferred to balance the overhead of packet headers. In contrast, as shown in Equations (4) and (5), a shorter packet is less vulnerable to packet loss due to staggered collisions with hidden terminals or channel fading errors. As a result, the effective throughput S_n does not necessarily monotonically increase as P_{SC} and P_E decease with a smaller packet size. Specifically, by combining Equations (6) through (8), we can numerically plot S_n as a function of packet size, L, as shown in Fig. 1. To generate the plot in Fig. 1, we assume $\tau_h = 0.01$, $P_{DC} = 0.1$, $P_E = 2E^{-5}$, and parameters T_{DIFS}, T_{SIFS}, T_{ACK}, T_{ACK_TO}, m, and m' are given by 802.11a specifications as shown in Table I.

TABLE I
SYSTEM PARAMETERS

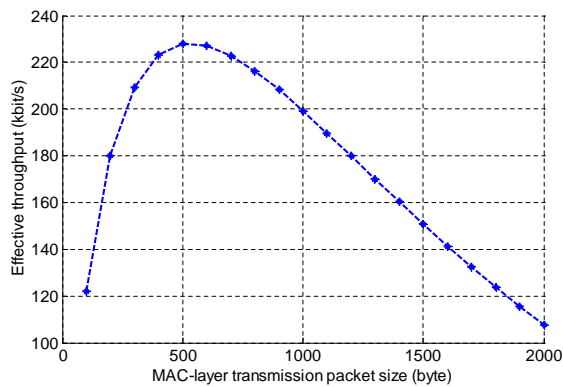| Parameter	Value	Parameter	Value
T_{SIFS}	16 μs	T_{DIFS}	34 μs
T_{ACK}	42.67 μs	T_{ACK_TO}	58.67 μs
m'	6	m	7
W	32	T_{slot}	9 μs
MAC header	246 bits	RTP/UDP/IP headers	320 bits

Fig. 1. Effective throughput S_n vs. MAC-layer transmission packet size L.

As seen in Fig. 1, for $L < 500$ bytes, the effective throughput S_n increases due to lower overhead of larger packet size. However, this trend reverses itself for $L > 500$ bytes, where the throughput decreases with increasing packet size. Specifically, for large values of L, the throughput loss due

TABLE II
SEARCH ALGORITHM FOR TRANSMISSION PACKET SIZE.

1: Initialize L_{min}, L_{max}, and L_1 with $L_{min} < L_1 < L_{max}$
2: $L \leftarrow L_1$ ▷ Apply L_1 for packetization
3: $L_{ind} \leftarrow 1$
4: **while** a new transmission occurs **do**
5:     Update number of Tx attempts $k_t \leftarrow k_t + 1$
6:     **if** $k_t \mod M_t == 0$ and $L_{max} - L_{min} > \varepsilon$ **then**
7:         **if** $L_{ind} == 1$ **then**
8:             Measure effective throughput $\rightarrow S_{n,1}$
9:         **else**
10:             Measure effective throughput $\rightarrow S_{n,2}$
11:         **end if**
12:         **if** $S_{n,2} == NULL$ **then**
13:             $L \leftarrow L_2 = L_1 + C \cdot (L_{max} - L_1)$
14:             $L_{ind} \leftarrow 2$ ▷ Apply L_2 for packetization
15:             Continue
16:         **end if**
17:         **if** $S_{n,2} > S_{n,1}$ **then** ▷ Apply golden section rule
18:             $L_{min} \leftarrow L_1; L_1 \leftarrow L_2$
19:             $L_2 \leftarrow L_1 + C \cdot (L_{max} - L_1)$
20:             $S_{n,1} \leftarrow S_{n,2}; L \leftarrow L_2$
21:         **else**
22:             $L_{max} \leftarrow L_2; L_2 \leftarrow L_1$
23:             $L_1 \leftarrow L_2 - C \cdot (L_2 - L_{min})$
24:             $S_{n,2} \leftarrow S_{n,1}; L \leftarrow L_1$
25:         **end if**
26:     **end if**
27: **end while**

to increased channel fading errors and staggered collisions is more pronounced than throughput gain due to reduced header overhead, resulting in a net decrease in S_n, as shown in Fig. 1. Thus, S_n is a unimodal function of L with only one maximum. Therefore, to maximize the effective throughput, the packet size should be large enough to balance the header overhead, and small enough to minimize the impact of both channel fading and staggered collisions.

B. Search Algorithm for Packet Size

As discussed in Section II-A, for WLANs with hidden terminals, the effective throughput is a unimodal function of the MAC-layer packet size. Thus, it is possible to use the relative increase or decrease of measured throughput to determine whether the packet size L needs to be increased or decreased in order to optimize throughput. For example, for the specific parameters resulting in Section II-A, if an increase in the packet size from L to $L + \Delta L$ leads to an increase in the throughput, then it can be concluded that $L_{opt} > L + \Delta L$, where L_{opt} is the optimal packet size maximizing the throughput. Conversely, if an increase in the packet size L to $L + \Delta L$ results in a decrease in the throughput, we conclude that $L_{opt} < L$. Therefore, it is possible to develop an iterative search algorithm based on throughput measurements in order to arrive at the optimal packet size. Specifically, we opt to use the golden section search [7] to

determine the optimal transmission packet size. The search algorithm maintains a triplet of points (L_{min}, L_1, L_{max}) in an iterative fashion, where $L_{min} < L_1 < L_{max}$ and their distances form a golden ratio. Initially, L_{min} is taken to be 50 bytes, and L_{max} is limited by the maximum transmission unit (MTU) size of 802.11 WLAN. That is, $L_{max} \leq L_{mtu}$, where $L_{mtu} = 2264$ bytes after excluding the MAC header and RTP/UDP/IP headers. First, L_1 is applied as the transmission packet size L. The node tracks the time duration to transmit M_t packets and counts the number of successful transmissions. As such, the effective throughput of the node can be measured by evaluating the number of bytes that are successfully transmitted over this time duration. Then, a new value $L_2 = L_1 + C \cdot (L_{max} - L_1)$ is chosen for the packet size L, where $C = \frac{3-\sqrt{5}}{2} \approx 0.38197$. The measured throughput corresponding to L_1 and L_2 are recorded as $S_{n,1}$ and $S_{n,2}$, respectively.

By comparing $S_{n,1}$ and $S_{n,2}$, a narrower size range is successively updated for $[L_{min}, L_{max}]$, inside which the maximum effective throughput is achieved. Table II shows the pseudo-code of the golden section search algorithm to determine an optimal packet size. As shown in lines (17 - 25), if $S_{n,1} < S_{n,2}$, we conclude that a larger throughput is achievable within (L_1, L_{max}), and update the size range by setting $L_{min} = L_1$. If $S_{n,1} > S_{n,2}$, the size range is narrowed down by setting $L_{max} = L_2$. The iterations continue until L_{min} and L_{max} converge to a tolerable level ε. Then, the converged size is applied to subsequent packet transmissions.

III. SIMULATION RESULTS

In this section, we present simulation results to evaluate the impact of packetization on network performance under various channel conditions and network configurations. Table I shows the simulation parameters following the specification of IEEE 802.11a. The channel rate is set at 12 Mbit/s. The node topology is shown in Fig. 2. As seen, the network consists of an AP, M nodes in the middle and K nodes that are distant from each other. The M nodes in the middle can sense all traffic, while the $\frac{K}{2}$ nodes at the left side can sense transmissions from all other nodes except the other $\frac{K}{2}$ nodes at the right side and vice versa. We refer to these K nodes as sensing-limited nodes. The search algorithm in Table II is applied to the K sensing-limited nodes to determine the packet size for MAC-layer transmissions. Since the nodes in the middle do not experience staggered collisions, our packetization strategy may not make a significant difference in their throughput. Hence, we focus on the throughput improvement of the K sensing-limited nodes. The total traffic load from all the associated nodes saturates the WLAN, in which the M nodes in the middle are sending background traffic at a constant rate.

First, we show an example of the search algorithm in Table II to determine the transmission packet size. Assume there are 4 sensing-limited nodes and 20 nodes in the middle sending constant-rate background traffic. The channel
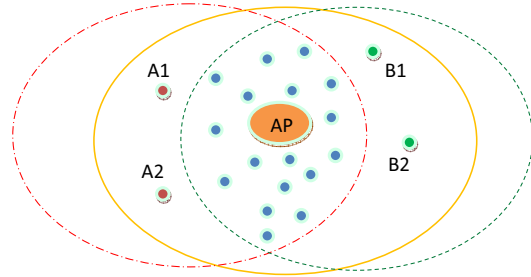

Fig. 2. Node topology for simulations.

BER is $2E^{-5}$. A sensing-limited node starts by setting $L_{min} = 50$ bytes and $L_{max} = 2000$ bytes. The node periodically measures its effective throughput for every $M_t = 400$ packet transmissions, which takes around 4 seconds. Then, according to the search algorithm, the size range $[L_{min}, L_{max}]$ is updated iteratively until it converges. As shown in Fig. 3, after 10 iterations, the packet size converges to around 664 bytes, which is used for subsequent MAC packetization. Fig. 4 shows the corresponding effective throughput as a function of system time.
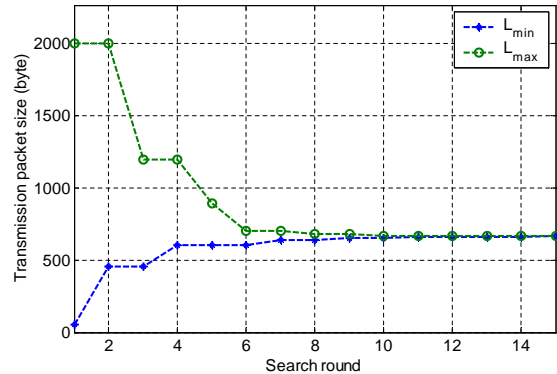

Fig. 3. Search for transmission packet size.

In Fig. 5, we compare the the effective throughput of sensing-limited nodes as a function of packet size L for different channel BER values. There are 4 sensing-limited nodes and 20 nodes in the middle sending constant-rate background traffic of about 120 kbit/s. The overall simulation time for packet transmission is around 30 minutes so that the network is stabilized for performance evaluation. In Fig. 5, the effective throughput for the searched packet size shown with a black circle marker is compared to those with fixed packet sizes ranging from 100 bytes to 1600 bytes in intervals of 200 bytes. In each curve, the searched packet size achieves the highest throughput, and its value decreases with a higher BER. This is because a smaller packet size is preferred to reduce packet loss due to increased channel fading errors. As seen, the effective throughput of sensing-limited nodes can be significantly improved by using the searched packet
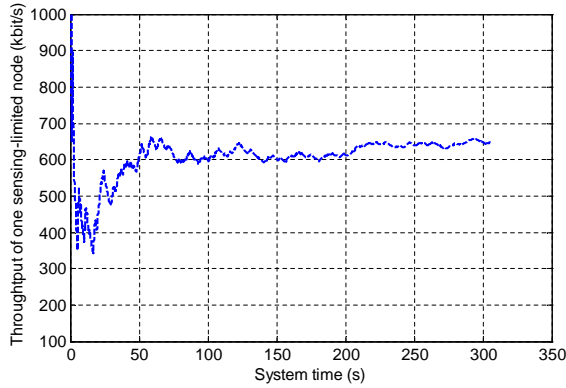
Fig. 4. Effective throughput of sensing-limited nodes as a function of time.

Fig. 6. Effective throughput of sensing-limited nodes as a function of packet size for various numbers of sensing-limited nodes.

size. For instance, when the channel BER is $2E^{-5}$, the effective throughput is increased by around 92% as compared to $L = 1600$ bytes.
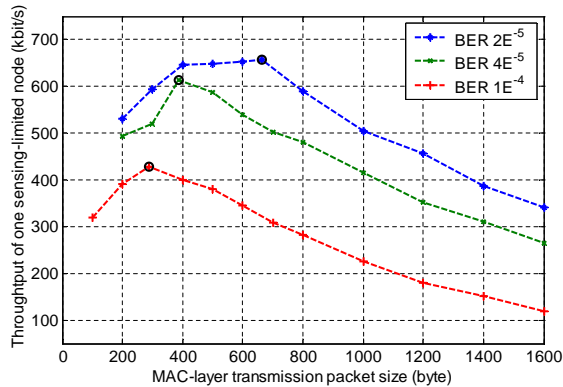

Fig. 5. Effective throughput of sensing-limited nodes as a function of packet size for various channel BER.

In Fig. 6, we repeat the simulations in Fig. 5 except that the channel BER is assumed to be $2E^{-5}$ and the number of sensing-limited nodes takes on values of 4, 8 and 12. It is observed that the packet size obtained by the search algorithm shown with a black circle marker decreases with the number of sensing-limited nodes. This is because more staggered collisions are introduced and a smaller packet size is required to mitigate the impact of staggered collisions. In [5], packet loss due to only channel fading and direct collisions are considered, and staggered collisions are not taken into account. Based on such an assumption, it is concluded that the optimal packet size does not vary with the number of contending nodes [5]. As seen in Fig. 6, this observation cannot be extended to the scenario with hidden terminals and staggered collisions. The packet size needs to be selected to be large enough to reduce header overhead and small enough to minimize packet loss due to both channel fading and staggered collisions. As seen in Fig. 6, the effective throughput of sensing-limited nodes can be significantly improved by using
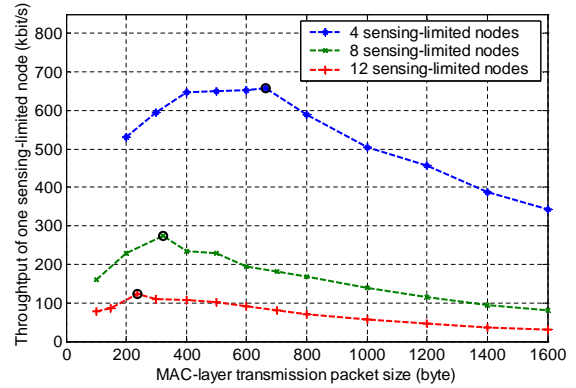
the searched packet size. For instance, when the number of hidden nodes is 12, the effective throughput is improved by around 213% as compared to $L = 1600$ bytes.

Fig. 7 shows the observed direct and staggered collision probabilities as a function of time, where the channel BER is $2E^{-5}$, and there are 12 sensing-limited nodes and 20 middle nodes sending constant-rate background traffic of about 120 kbit/s. As seen, the selected packet size significantly reduces the staggered collision probability, which in turn increases the effective throughput. This is at the expense of some increase in the direct collision probability though.
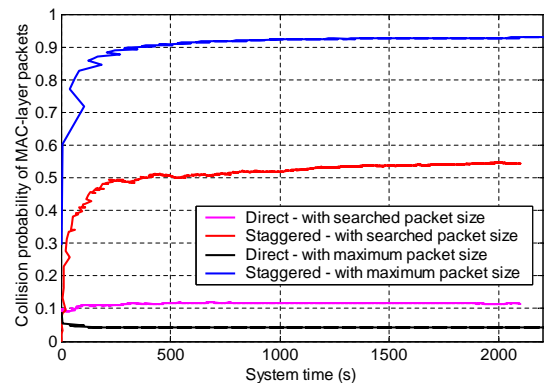

Fig. 7. Collision probabilities of sensing-limited nodes with searched packet size and maximum packet size L_{mtu}.

In Fig. 8, we show the effective throughput of sensing-limited nodes as a function of the sending rate of background traffic from 20 middle nodes. It is assumed that the channel BER is $2E^{-5}$, and there are 4 sensing-limited nodes. The effective throughput with the searched packet size is compared to that with $L_{mtu} = 2264$ bytes. It is observed that the effective throughput is improved more for lower background traffic load. For instance, when the middle nodes are transmitting at a rate of 200 kbit/s, the effective throughput increases by 146% from 109 kbit/s to 268 kbit/s. In contrast, when the
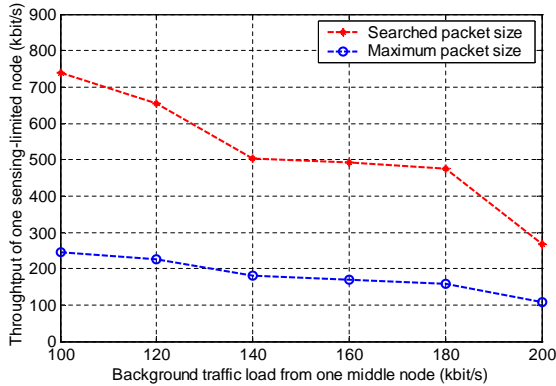
Fig. 8. Effective throughput of sensing-limited nodes as a function of background traffic load from middle nodes with searched packet size and maximum packet size L_{mtu}.
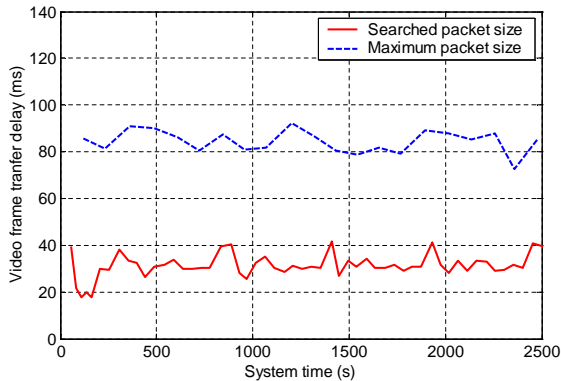

Fig. 9. Average video frame transfer delay with searched packet size and maximum packet size L_{mtu}.

sending rate of middle nodes is 100 kbit/s, there is a much larger throughput improvement of 199% from 246 kbit/s to 737 kbit/s. This is because for lower level of background traffic, the network is less saturated and there is more room for the packetization algorithm to take effect before saturation.

Fig. 9 shows the average transfer delay of video frames from one sensing-limited node as a function of time. The channel BER is assumed to be $1E^{-4}$. There are 4 sensing-limited nodes transmitting an H.264-coded video sequence *NBC 12 News* at CIF resolution, 30 frames/s, and a GoP size of 16 with a GoP structure of "$IB_1B_2B_3P_1B_4B_5B_6P_2B_7B_8P_3B_9B_{10}B_{11}B_{12}$" [8]. This results in an average video coding rate of 497 kbit/s. If the size of a video I, P, or B frame is larger than the MAC-layer packet size limit L, the video frame is fragmented and transmitted through multiple MAC-layer packets. Also, multiple video frames can be aggregated into one MAC-layer packet if their total size is no greater than the packet size limit L. However, no P and B frames are split across multiple MAC-layer packets. Fig. 9 shows the transmission duration of either the successful delivery of a video frame by the MAC layer or

failure to deliver after reaching the retransmit limit. Each point on the curves corresponds to the transfer delay averaged over consecutive 1400 video frames for a given node. As seen, the packet size determined by our search algorithm reduces the average video frame transfer delay from 84 ms to 31 ms. For interactive video applications with 150 ms delay bound, a reduction of 53 ms is quite significant. This reduced delay can be intuitively explained by considering that the optimal packet size is usually significantly smaller than the maximum packet size, thus resulting in fewer collisions, fewer retransmissions, and hence lower delay.

IV. CONCLUSION AND FUTURE WORK

In this paper, we have analyzed the impact of channel fading, and direct collisions and staggered collisions on MAC-layer packetization in 802.11 WLANs. On one hand, a large packet size is preferred to minimize protocol header overhead. On the other hand, in the presence of hidden terminals, the transmission packets need to be small enough to reduce packet loss due to both channel fading and staggered collisions. Based on measured throughput and golden section search, we have developed an iterative algorithm to search for the optimal packet size in order to achieve a reasonable trade-off between minimizing header overhead, fading errors, staggered collisions. Our work goes beyond that in [5] in that we consider staggered collisions as well as fading and direct collisions. It is observed that the searched packet size depends on the channel BER, the number of sensing-limited nodes, and the background traffic load level. By applying the searched packet size for MAC-layer packetization, we can not only improve the effective throughput but also reduce video frame transfer delay. Future work includes verification of our results through NS-2 simulations, and incorporating forward error control in our analysis.

REFERENCES

[1] M. Krishnan, S. Pollin, and A. Zakhor, "Local estimation of collision probabilities in 802.11 WLANs with hidden terminals," Technical Report, EECS Department, University of California Berkeley, Jan. 2009.

[2] S. Wenger, "H.264/AVC over IP," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 645–656, July 2003.

[3] M. van der Schaar and D. S. Turaga, "Cross-layer packetization and retransmission strategies for delay-sensitive wireless multimedia transmission," *IEEE Trans. Multimedia*, vol. 9, no. 1, pp. 185–197, Jan. 2007.

[4] D. Qiao, S. Choi, and K. G. Shin, "Goodput analysis and link adaptation for IEEE 802.11a wireless LAN," *IEEE Trans. Mobile Comput.*, vol. 1, no. 4, pp. 278–292, Oct.-Dec. 2002.

[5] J. Yin, X. Wang, and D. P. Agrawal, "Optimal packet size in error-prone channel for IEEE 802.11 distributed coordination function," in *Proc. IEEE WCNC*, vol. 3, Mar. 2004, pp. 1654–1659.

[6] L. X. Cai, X. Shen, J. W. Mark, L. Cai, and Y. Xiao, "Voice capacity analysis of WLAN with unbalanced traffic," *IEEE Trans. Veh. Technol.*, vol. 55, no. 3, pp. 752–761, May 2006.

[7] W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes in C: The Art of Scientific Computing (2nd Edition)*. Cambridge University Press, Oct. 1992.

[8] P. Seeling, F. H. Fitzek, and M. Reisslein, *Video Traces for Network Performance Evaluation*. Springer, Nov. 2006.