Formulating and Evaluating Impacts of IPv6 Header Overhead for Videos over Wireless Networks

Chu-Chuan Lee [2], Wen-Hsien Chu [1], and Pao-Chi Chang [1]

[1] *Department of Communication Engineering, National Central University, Taiwan, R.O.C.*
[2] *Chunghwa Telecom Co., Ltd., Taiwan, R.O.C.*
E-mail: { cclee, pcchang }@vaplab.ee.ncu.edu.tw

Abstract

IPv6 is designed to satisfy the dramatically growing number of users and multimedia applications in the current and next generation networks. Using the WLAN technology can effectively extend the IPv6 world from wired to wireless infrastructures. Although IPv6 protocol belongs to the Network Layer, the header increase of IPv6 may affect the effective throughput of WLAN and the video coding work. This paper formulates a simply yet robust closed-form for finding an IPv6 packet size that can maximize the WLAN throughput based on the header overhead of IPv6 packets, the wireless channel condition with burst bit errors, and the finite retransmission number of ARQ (Automatic Repeat Request). The formulated closed-form can: 1) accurately and timely determine the most suitable IPv6 packet size according to the current wireless channel status; 2) effectively quantify the WLAN throughput degradation due to the header increase of IPv6. This study also explores the impact of header overhead of IPv6 packets to the video coding work and proposes a Header Overhead Accommodation (HOA) scheme to solve the problem. Analytical and simulation results verify the accuracy and effectiveness of the proposed closed-form and HOA methods. The contributions are robust for various IPv6 implementations such as the IPSec (IP Security) over different 802.11 standards.

Keywords*: IPv6, multimedia service, wireless network*

I. Introduction

There is a strong consensus today that IP will be the foundation of next-generation networking [1]. The continuous growth of the Internet world requires that its overall architecture can evolve to accommodate new technologies for satisfying the growing numbers of users, applications, and devices. IPv6 is designed to satisfy these requirements and allow the return to a global end-to-end environment where the addressing rules of network are transparent to applications again [2]. IPv6 quadruples the number of network address bits from 32bits of IPv4 to 128bits, which provides enough globally unique IP addresses for every network device on the planet. In addition to unlimited IP addresses, IPv6 also provides a number of advances, including: 1)Neighbor Discovery (ND) and auto-configuration functions which can provide the plug and play capability; 2) end-to-end security by means of IP Security (IPSec) technique. The IPSec architecture is mandatory for IPv6 implementation, but not required for IPv4; 3) better support of IP mobility by means of Mobile IPv6 protocol; 4) enhanced QoS capability with the Traffic Class field and the Flow Label field; 5) flexible header extension that follows the main IPv6 header; 6) improved multicasting routing with the multicast address being extended by a scope field [3][4].

The support of IEEE 802.11-based Wireless Local Area Networks (WLANs) as an extension to existing wired infrastructure is important for IPv6 development. With low cost and high-speed data rate capabilities, the popularity of WLANs is growing exponentially. There are three IEEE standards available: the Complementary Code Keying (CCK)-based 802.11b [5] in the 2.4 GHz band, the Orthogonal Frequency Division Multiplex (OFDM)-based 802.11a [6] in the 5 GHz band, and the 802.11g [7] based on the same OFDM technology as employed in 802.11a. The 802.11b has the advantage of worldwide spectrum availability while the 802.11a is capable of supporting more high data rate schemes. The 802.11g is formally ratified in June 2003, plus backward compatibility with 802.11b devices. With the substantial increase in the bit rate available such as 802.11a or 802.11g, multimedia applications over WLANs become a reality and thus require the sufficient support of IP addresses.

Although wireless networks provide users an easy way for accessing audio/video (AV) contents, the received quality is easily affected due to the co-channel interference, multi-path fading, competing traffic, mobility, etc. In wireless networks, ARQ (Automatic Repeat Request) protocols are widely used for error control because they are simple and provide high system reliability. However, the permitted retransmission number of ARQ is strictly limited due to the tight delay constraint of AV services. On the other hand, an adequate packetization mechanism also can improve the received picture quality in wireless networks. However, a packetization mechanism using the long packet size for video data may worsen the packet error rate (PER) at the

client side if the wireless channel condition is poor. In such a case, the effective bandwidth utilization decreases and the received picture quality degrades significantly. In contrast, when the packetization mechanism adopts a short packet size to packetize the video data, the increased header overhead also decreases the bandwidth utilization. This phenomenon is a tradeoff and is particularly true in the IPv6 network since the header size of IPv6 packets is larger than that of IPv4 packets. When the demands and technologies of IPv6 are gradually mature, the quantitative analyses to the influence of header overhead of IPv6 packets for videos over WLANs and the corresponding solutions that can effectively reduce the influences are important.

Currently, there are many research results in this area. Regarding performances of IPv6, Ariga et al. [8] evaluated the performance of IPv6 data transmission using IPSec (IP Security). In [9], the test of maximum IPv6 forwarding rate is executed and compared with the result using IPv4 protocol. Raicu and Zeadally [10][11] constructed two IPv6 testbeds for obtaining the throughput and latency results of TCP and UDP traffics. The results of [8]-[11] are valuable, however, the impact of IPv6 header overhead to the effective throughput of WLAN and to the video coding work are not discussed. Schwartz [12] evaluated the throughput efficiency for a stop-and-wait protocol such as the ARQ procedure in wireless networks. Based on [12], Modiano [13] further proposed an adaptive packetization algorithm for wireless networks with the ARQ protocol. They utilized the random bit error model and assumed the permitted retransmission number of ARQ to be infinite for simplifying the analysis process. However, the assumption of infinite ARQ retransmissions is not suitable for delay-sensitive AV contents and the use of random bit error model is not proper to a wireless channel with burst bit errors. Besides, there are two compression protocols for IPv6/IPv4 header emerged from the Internet Engineering Task Force (IETF), i.e., the Internet Protocol Header Compression (IPHC) [14][15] and the Robust Header Compression (ROHC) [16][17]. IPHC is suitable for links with low bit error rate (BER) and ROHC is designed for wireless links with high BER. However, high implementation complexity and rare commercial application are the main bottlenecks of ROHC. In addition, regardless of IPHC or ROHC, the header compression requires extra resources on nodes that instantiate the compression algorithm. Therefore, both header compression methods are not included in this paper.

In this paper, we address the IPv6 packetization problems for video contents over WLANs. This study targets at the PCF (Point Coordination Function) mode of WLAN since it is more appropriate for delay-sensitive AV applications. Considering the header of IPv6 packets, the wireless channel condition with burst bit errors, and the finite retransmission number of ARQ, we formulate a simple yet robust closed-form for finding an IPv6 packet size that can maximize the effective throughput of WLAN. By comparing these throughput results using different IPv6 packet sizes, the throughput degradation is minimum if the most suitable IPv6 packet size is used. In this paper, an IPv6 packet includes the payload of video data and headers of Transport Layer and Network Layer. Moreover, we observe that the header overhead of IPv6 packets also may affect the estimation of target encoding bits during the video coding work. To solve the problem, we propose a Header Overhead Accommodation (HOA) scheme and automatically consider the variation of header overhead in the video coding process.

This paper is constructed as follows. In Section II, this study gives an overview of related background information, including the IPv6 header structure and the PCF mode of 802.11b/a/g networks. In Section III, we present the theoretical analyses for achieving the maximum effective throughput of 802.11b/a/g/e networks. A closed-form of the most suitable IPv6 packet size is formulated. In Section IV, we propose a method that can automatically adjust the video encoding bit rate in the video application layer for accommodating the variation of header overhead. In Section V, we validate the numerical results by comparing with simulation results. Finally, Section VI concludes this paper.

II. Backgrounds

Regarding the IP header structure, the minimum header length of IPv4 is 20 bytes, but if options are added, it can be extended in 4-byte increments up to 60 bytes. On the other hand, the IPv6 header has a fixed length of 40 bytes followed by extension headers. The detailed structure of IPv6 header is specified in RFC 2460 [18], as shown in Fig. 1. The current IPv6 specification also defines six extension headers, i.e., Hop-by-Hop Options header, Routing header, Fragment header, Destination Options header, Authentication header, Encrypted Security Payload header. Theoretically, the added header overhead due to the IPv6 header expansion is only 1.318 percent compared with the Etherent maximum transmission unit (MTU) size of 1518 bytes. However, the actual throughput degradation of WLAN due to the header increase of IPv6 is larger than the above expectation. The main reason is that a long header length increases the PER in WLANs. Moreover, considering the IPv6 implementation with IPSec, there are two basic security protocols: the Authentication Header (AH) and the Encapsulating Security Payload (ESP). Each protocol can operation in one of two modes: transport mode and tunnel mode. In the tunnel mode, the entire original datagram is regarded as the payload, and a newly created outer IPv6 header and an AH header are included [19].

Clearly, the header overhead of network layer increases significantly when IPSec is used.

With respect to WLAN techniques, there are two access methods in the 802.11 MAC (Medium Access Control) layer: the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF) [20]. The DCF mode is the basic access mechanism that uses the carrier sense multiple access/collision avoidance (CSMA/CA) scheme and the PCF mode is designed for supporting applications that require real-time and contention-free delivery. In PCF mode, Point Coordinator (PC) centrally controls the access to the wireless medium. When transmitting the data, PC appropriately schedules the downlink traffic for delivery to different wireless stations and grants the stations access to the medium through a polling mechanism for uplink traffic. Due to the contention-free characteristic, PCF mode is particularly suitable for delay-sensitive AV applications.

Fig.2 shows various data transmission cases between PC and wireless stations in the PCF mode. At the beginning of the Contention-Free Period (CFP), PC transmits a Beacon frame. One important component of the beacon announcement is the maximum duration of CFP, i.e., CFPMaxDuration. During CFP, stations may transmit only if PC solicits the transmission with a polling frame that is abbreviated as CF-Poll. All contention-free transmissions are separated by the Short InterFrame Space (SIFS) that is the shortest space among various interframe spaces or separated by the PCF Interframe Space (PIFS) that is used in the PCF mode. As with traditional Ethernet, these interframe spaces play an important role in coordinating access to the transmission medium. These default values of 802.11b/a/g parameters are presented in Table I.

III. Formulating the IPv6 packet size

For formulating the closed-form of the most suitable IPv6 packet size in WLAN environments, this work uses the Gilbert model [21] to characterize the error sequences generated by wireless transmission channel, as shown in Fig.3. From [21] the average bit error rate p_{be} and the average burst error length b of Gilbert model can be derived as

$$p_{be} = \frac{(1-\beta)\cdot p_B + (1-\alpha)\cdot p_G}{(1-\alpha)+(1-\beta)} \tag{1}$$

$$b = \frac{1}{(1-\alpha)} \tag{2}$$

where p_G and p_B represent the probabilities that bit errors occur in the Good state (G) and the Bad state (B), respectively. Meanwhile, considering a video packet with P bits of encoded video data and H_{234} bits of cross layer headers from Transport Layer to Data Link Layer, the average PER p_{pe} in a wireless channel that has burst bit errors is formulated by this work as

$$p_{pe} = 1-\left\{ \left[P(G|G)\cdot P(G)+P(G|B)\cdot P(B)\right] \cdot \beta^{P+H_{234}-1} \right\} \tag{3}$$

By substituting (1) and (2) into (3) and evaluating (3), we have

$$\begin{aligned} p_{pe} &= 1-\left\{ \left[\beta\cdot\frac{(1-\alpha)}{(1-\alpha)+(1-\beta)} + (1-\alpha)\cdot\frac{(1-\beta)}{(1-\alpha)+(1-\beta)} \right] \cdot \beta^{P+H_{234}-1} \right\} \\ &= 1-\left\{ (1-p_{be})\cdot\left[1-\frac{p_{be}}{(1-p_{be})\cdot b} \right]^{P+H_{234}-1} \right\} \end{aligned} \tag{4}$$

where the values of p_G and p_B of (1) are set to 0 and 1 based on [21], respectively. Therefore, the probability that a video packet is successfully delivered to the client after n retransmissions in the WLAN network is given by

$$p_s = (1-p_{pe})\cdot(p_{pe})^n \tag{5}$$

If L_r is the permitted maximum retransmission number for a video packet over a wireless channel with p_{pe}, the probability that the packet is successfully delivered within L_r retransmission limit is computed as

$$\begin{aligned} p_{s\_L_r} &= (1-p_{pe}) + (1-p_{pe})\cdot p_{pe} + \cdots + (1-p_{pe})\cdot p_{pe}^{L_r} \\ &= (1-p_{pe})\cdot \sum_{k=1}^{L_r+1} p_{pe}^{k-1} \\ &= (1-p_{pe}^{L_r+1}) \end{aligned} \tag{6}$$

In other words, the probability that the delivery for the packet is not successful after L_r retransmission limit is given by

$$p_{f\_L_r} = p_{pe}^{L_r+1} \tag{7}$$

Consequently, the mean transmission number that a video packet is successfully delivered within L_r retransmission limit is calculated by

$$\begin{aligned} S_{succ} &= 1\cdot\frac{(1-p_{pe})}{P_{s\_L_r}} + 2\cdot\frac{p_{pe}\cdot(1-p_{pe})}{P_{s\_L_r}} + \cdots + (L_r+1)\cdot\frac{p_{pe}^{L_r}\cdot(1-p_{pe})}{P_{s\_L_r}} \\ &= \frac{1}{P_{s\_L_r}}\cdot\left\{ 1\cdot\left[(1-p_{pe})\right] + 2\cdot\left[p_{pe}\cdot(1-p_{pe})\right] + \cdots + (L_r+1)\cdot\left[p_{pe}^{L_r}\cdot(1-p_{pe})\right] \right\} \\ &= \left(\frac{1}{1-p_{pe}}\right) - \left[\frac{(L_r+1)\cdot p_{pe}^{(L_r+1)}}{1-p_{pe}^{(L_r+1)}}\right] \end{aligned} \tag{8}$$

Then, the mean transmission number for a video packet with the limit of L_r retransmissions is then given by

$$\begin{aligned} S_{avg} &= \left[S_{fail}\times p_{f\_L_r} \right] + \left[S_{succ}\times p_{s\_L_r} \right] \\ &= \left[(L_r+1)\times p_{f\_L_r} \right] + \left[S_{succ}\times p_{s\_L_r} \right] \end{aligned} \tag{9}$$

Based on Fig.2, the duration of a successful transmission, i.e., neither the video data packet nor the ACK packet is in error, can be computed as

$$T_{s\_cyc} = T_{data} + 2 \cdot T_{SIFS} + T_{ack} \tag{10}$$

where

$$T_{data} \simeq \left(\frac{P + H_{234}}{R} \right) + T_P \tag{11}$$

$$T_{ack} \simeq \left(\frac{8 \times 14}{R} \right) + T_P \tag{12}$$

where R is the data rate of WLAN, T_P is the required time that transmits the preamble bits and the header of WLAN physical layer, and the length of an ACK packet is 14 bytes.

Similarly, the duration of a failure transmission i.e., either the video data packet or the ACK packet is in error, can be calculated by

$$T_{f\_cyc} = T_{data} + T_{PIFS} \tag{13}$$

where the duration of a failure transmission due to an erroneous ACK packet is assumed to be the same as that due to an erroneous video data packet for simplifying the analysis. The difference between the two failure transmission durations is very little.

Assuming that the delivery of video packets is a sequence of independent Bernoulli trials, the expected number of Bernoulli trials until the first successful video packet is received by the wireless station is just the reciprocal of (6) and expressed as

$$N_p = \frac{1}{p_{s\_L_r}} = \frac{1}{(1 - p_{pe}^{L_r+1})} \tag{14}$$

Now, from the station viewpoint, the average time interval between two correctly received video packets can be computed by

$$\begin{aligned} T_{total} &= T_{s\_cyc} + \left(N_p \cdot S_{avg} - 1 \right) \cdot T_{f\_cyc} \\ &= N_p \cdot S_{avg} \cdot \left(\frac{P + H_{234}}{R} + T_P + T_{PIFS} \right) + 2 \cdot T_{SIFS} + T_{ack} - T_{PIFS} \end{aligned} \tag{15}$$

By substituting (4), (9) and (14) into (15) and evaluating (15), we have

$$\begin{aligned} &T_{total} \\ &= \frac{1}{\beta^{P+H_{234}-1}} \left[\begin{array}{l} \dfrac{1}{1 - p_{be}} \left(\dfrac{P + H_{234}}{R} + T_P + T_{PIFS} \right) + \\ \beta^{P+H_{234}-1} \left(2T_{SIFS} + T_{ack} - T_{PIFS} \right) \end{array} \right] \end{aligned} \tag{16}$$

where

$$\beta = \left(1 - \frac{p_{be}}{(1 - p_{be}) \cdot b} \right) \tag{17}$$

In the saturated transmission case, the maximum throughput of WLAN, in packets/sec delivered, is just the reciprocal of T_{total}. The effective throughput D_e, in bits/sec delivered, is then given by

$$\begin{aligned} D_e &= \frac{1}{T_{total}} \cdot P \\ &= \frac{P \cdot \beta^{P+H_{234}-1}}{\dfrac{1}{1 - p_{be}} \left(\dfrac{P + H_{234}}{R} + T_P + T_{PIFS} \right) + \beta^{P+H_{234}-1} \cdot \left(2T_{SIFS} + T_{ack} - T_{PIFS} \right)} \\ &\triangleq \frac{f(P)}{g(P)} \end{aligned} \tag{18}$$

By differentiating Eq. (18) with respect to P and setting the derivative to 0, we have

$$f(P) g'(P) = f'(P) g(P) \tag{19}$$

where

$$f(P) = P \cdot \beta^{P+H_{234}-1} \tag{20}$$

$$f'(P) = \beta^{P+H_{234}-1} + P \cdot \beta^{P+H_{234}-1} \cdot \ln \beta \tag{21}$$

$$\begin{aligned} g(P) = &\frac{1}{1 - p_{be}} \cdot \left(\frac{P + H_{234}}{R} + T_P + T_{PIFS} \right) + \\ &\beta^{P+H_{234}-1} \cdot \left(2 \cdot T_{SIFS} + T_{ack} - T_{PIFS} \right) \end{aligned} \tag{22}$$

$$\begin{aligned} g'(P) = &\frac{1}{R \cdot (1 - p_{be})} + \\ &\beta^{P+H_{234}-1} \cdot \left(2 \cdot T_{SIFS} + T_{ack} - T_{PIFS} \right) \cdot \ln \beta \end{aligned} \tag{23}$$

By substituting (20) - (23) into (19) and evaluating (19), we have

$$\begin{aligned} &\frac{P}{R \cdot (1 - p_{be})} + \beta^{P+H_{234}-1} \cdot P \cdot \left(2T_{SIFS} + T_{ack} - T_{PIFS} \right) \cdot \ln \beta \\ &= \left(1 + P \cdot \ln \beta \right) \left[\begin{array}{l} \dfrac{1}{1 - p_{be}} \left(\dfrac{P + H_{234}}{R} + T_P + T_{PIFS} \right) + \\ \beta^{P+H_{234}-1} \cdot \left(2T_{SIFS} + T_{ack} - T_{PIFS} \right) \end{array} \right] \end{aligned} \tag{24}$$

After further evaluating (24), we obtain

$$C_2 \cdot \beta^P + \ln \beta \cdot P^2 + C_1 \cdot \ln \beta \cdot P + C_1 = 0 \tag{25}$$

where

$$C_1 = H_{234} + R \cdot \left(T_P + T_{PIFS} \right) \tag{26}$$

$$C_2 = \beta^{H_{234}-1} \cdot R \cdot (1 - p_{be}) \cdot \left(2 \cdot T_{SIFS} + T_{ack} - T_{PIFS} \right) \tag{27}$$

Meanwhile, we extend the item β^P of (25) as

$$\beta^P = \left(1 - \frac{p_{be}}{(1-p_{be})\cdot b}\right)^P = (1-C_3)^P$$
$$= 1 - C_3 \cdot P + C_3^2 \cdot \frac{P!}{2!(P-2)!} - C_3^3 \cdot \frac{P!}{3!(P-3)!} + ... \quad (28)$$
$$= 1 - C_3 \cdot P + C_3^2 \cdot \frac{1}{2} \cdot P(P-1) - C_3^3 \cdot \frac{1}{6} \cdot P(P-1)(P-2) + ...$$

where $\beta = 1 - C_3$. For simplifying the analyses and obtaining a closed-form, we ignore the later items of (28) whose power of P exceeds two. Then, Eq. (28) can be simplified as

$$\beta^P \simeq 1 - C_3 \cdot P + C_3^2 \cdot \frac{1}{2} \cdot P(P-1) \quad (29)$$

By substituting (29) into (25) and evaluating (25), we have

$$\left[\frac{1}{2}C_3^2 \cdot C_2 + \ln\beta\right] \cdot P^2 + \left[C_1 \ln\beta - C_2\left(C_3 + \frac{1}{2}C_3^2\right)\right] \cdot P + (C_1 + C_2) = 0 \quad (30)$$

Finally, the optimal value of *P* that maximizes the effective throughput D_e can be calculated from (30) and is given as

$$P_{opt} = \frac{(C_4 - C_1 \cdot \ln\beta) - \sqrt{(C_1 \cdot \ln\beta - C_4)^2 - 2[(1-\beta)^2 \cdot C_2 + 2\ln\beta] \cdot (C_1 + C_2)}}{(1-\beta)^2 \cdot C_2 + 2\ln\beta} \quad (31)$$

where

$$C_4 = C_2\left(C_3 + \frac{1}{2}C_3^2\right) = \frac{C_2 \cdot (\beta^2 - 4\beta + 3)}{2} \quad (32)$$

After adding the headers of cross layers to (31), we can determine the most suitable IPv6 packet size for any wireless channel condition easily and timely. The contribution can be applied directly to an adaptive packetization mechanism where a dynamic IPv6 packet size is used for coping with the variable wireless channel condition. Moreover, regarding different IPv6 implementations such as IPSec, tunneling mechanisms, etc., we also can easily obtain the impact of header overhead of IPv6 packets to WLAN throughput by means of (31).

IV. Accommodation of header overhead

In the traditional video coding process, the estimation of target encoding bits is based on the available network bandwidth. In general, the header overhead of cross layers is not taken into consideration in the estimation procedure. This may cause the mismatch between the available network bandwidth and the actual bandwidth demand of the encoded video. Besides, when an adaptive packetization mechanism is used, the header overhead is variable due to the dynamic packet size strategy. The variable header overhead may further worsen the bandwidth mismatch problem and thus increase the complexity of resource management. To solve these problems, this work proposes the HOA scheme, which is executed in the video application layer, to adjust automatically the video encoding bits based on the current variation of header overhead.

In this section, a discrete time model with the unit of frame number is used. At the beginning of the video coding process, HOA still sets the target encoding bits to the available network bandwidth directly. When the video encoder finishes all encoding operations for the current video frame f_n at time *n*, HOA computes the required header overhead H_n for packetizing the encoded bits of f_n by

$$H_n = H_{234} \times \left\lceil \frac{A_n}{P_n} \right\rceil \quad (33)$$

where A_n is the encoded bits of f_n and P_n is the calculated payload length by (31). Clearly, the value of H_n increases if a small payload length is used or the value of H_{234} is enlarged. In addition, the total header overheads cumulated from the same video sequence under various wireless channel conditions are also different if the strategy of dynamic packet sizes is used.

Now, HOA takes the header overhead H_n of f_n into consideration while encoding the next video frame f_{n+1}. When the encoding procedure for f_{n+1} begins, the target encoding bits for f_{n+1} is estimated by the following improved expression

$$R_{n+1} = \frac{B_{n+1} - H_n}{u_{n+1}} \cdot (1 - S) + A_n \cdot S \quad (34)$$

where

B_{n+1} : remaining available bits for the video sequence at time *n*+1.

R_{n+1} : target encoding bits for the video frame f_{n+1}.

u_{n+1} : remaining raw video frames which are not encoded yet at time *n*+1.

S : weighting factor to determine the impact of the previous frame on the calculation of target encoding bits. The value is set to 0.05 based on the MPEG-4 standard [22].

From (34), the header overhead H_n of f_n is subtracted from the remaining available bits B_{n+1} before determining the target encoding bits of f_{n+1}. Repeating the procedure, the required header overhead of each video frame is automatically accommodated into the budget of

available encoding bits of the video sequence.

V. Numerical results and discussions

This study constructs two WLAN simulation environments, 802.11g and 802.11b, to verify the accuracy of the formulated closed-form and to evaluate the impact of header overhead of IPv6 packets to WLAN throughput. This work assumes that the OFDM technique is used in the physical layer of 802.11g and the maximum bandwidth of 802.11g is set to 12 Mbps. The 802.11a network is not discussed here since its performances and parameters are similar to the case of 802.11g with OFDM technique. We use the Gilbert model with ten BERs and three burst error lengths to simulate various wireless channel conditions.

Fig. 4 shows 802.11g throughputs versus IPv6 packet sizes under different BERs and burst error lengths. Regarding the case x-y, x = 1, 2, and 3, represent the BER to be 4×E-4, 1×E-4, and 8×E-5, respectively. In addition, y = 1 and 2 represent the burst error length to be 3 and 7, respectively. From the simulation results of each curve, we can find an IPv6 packet size that has the maximum throughput. On the other hand, we can directly use the proposed closed-form to obtain the analytically IPv6 packet size for each wireless channel condition. These analytical results are plotted by black dots in Fig. 4. Clearly, the analytical and simulated results are in good agreement at each wireless channel condition. This verifies the accuracy of the closed-form formulated by this paper.

From Fig. 4 we also observe that, when the wireless channel condition is poor, the throughput degradation due to the packet error of a long packet size is significantly larger than that due to the header overhead of IPv6 packets in a short packet size. However, while the wireless channel condition is toward good status, the header overhead of IPv6 packets obviously affects the throughput if an inadequate packet size is used. Besides, under the same BER, a short burst error length also causes larger throughput degradation than a long burst error length. Fig. 5 uses the same simulation conditions except that 802.11g is replaced by 802.11b. From Fig. 5 we obtain similar conclusions among the IPv6 packet size, the BER and the throughput. These results of Fig. 5 also exhibit that the proposed closed-form is robust for various 802.11 WLAN standards.

Although both IPv4 and IPv6 header overheads degrade the effective throughput of WLAN, the grade of throughput degradation is different. We evaluate the WLAN throughput degradation due to the header increase of IPv6 under various wireless channel conditions, as shown in Fig. 6. The analytically IPv4 and IPv6 packet sizes are used for each combination of BER and burst error length. The throughput difference in Fig. 6 is defined as $[(Thr_{v4} - Thr_{v6})/ Thr_{v4}] \times 100\%$, where $Thr_{v\#}$ represents the maximum throughput of 802.11g using the proposed IPv# packet size. From Fig. 6, the throughput difference is close to a constant if the wireless channel condition is good. However, the throughput difference increases if the wireless channel condition is toward poor status. We observe that the throughput difference increases up to 20.4% when the BER is equal to 1×E-3 with 1-bit burst error length. The main reason is that a short packet size is used in the poor wireless channel condition, and the header overhead of IPv6 is larger than that of IPv4 when a short packet size is used to packetize the data. Fig. 7 uses the same simulation conditions except that 802.11g is replaced by 802.11b. From Fig. 7, we obtain similar summaries and observe that the throughput difference increases up to 16.5%.

To understand the maximum throughputs of WLAN under various IPv6 implementations, we further evaluate the impact of header overhead of IPv6 IPSec packets to 802.11g and 802.11b throughputs with different BERs and burst error lengths, as shown in Figs.8 and 9. As mentioned in Section II, the header overhead of network layer increases significantly when IPSec is used. In this simulation scenario, we consider the AH protocol with tunnel mode. In the tunnel mode, a newly created outer IPv6 header and an AH header are added to the original data datagram. By comparing the results of Figs.8 and 9 with that of Figs.4 and 5, we find that the throughput degrades up to 0.6Mbps in the case 1-1 of 802.11g due to the increase of IPv6 IPSec header.

Finally, this work explores the bandwidth mismatch problem, as presented in Table II. This paper uses the MPEG-4 video codec [22] to generate a standard test sequence “Foreman” and uses four different video packet sizes for discussion. The test sequence is the CIF format in which a GOP consists of 15 frames with IBBP pattern. The available network bandwidth is assumed 1Mbps and the target encoding bit rate is set to 1Mbps based on the available network bandwidth directly. Meanwhile, considering the packetization work without HOA, the difference between the available network bandwidth and the actual bandwidth requirement for delivering Foreman is significant, especially in the case of short packet size. This phenomenon is mainly due to the header overhead. Fortunately, when the proposed HOA method is activated, the actual bandwidth requirement is very close to the available network bandwidth, regardless of the used video packet sizes. Notably, form (34) the header overhead of each video packet is automatically and accurately accommodated into the video coding process. No coarse bandwidth reservation for header overhead is required in the stage of determining the target encoding bit rate.

VI. Conclusion

Although IPv6 protocol belongs to the Network Layer, the header increase of IPv6 still affects the WLAN throughput and the video coding work. This paper formulates a simple yet robust closed-form that can: 1)

accurately and timely determine the most suitable IPv6 packet size based on the current wireless channel condition; 2) effectively quantify the impact of header overhead of IPv6 packets to WLAN throughput. Besides, the proposed HOA mechanism can automatically and accurately accommodate the header overhead of each video packet into the video coding process. The contributions are robust for various IPv6 implementations such as the IPSec over different 802.11 standards.

References

[1] S. I. Maniatis, E. G. Nikolouzou, and I. S. Venieris, "End-to-end QoS specification issues in the converged all-IP wired and wireless environment," IEEE Commun. Mag., vol. 42, pp. 80-86, June 2004.

[2] A. Durand, "Deploying IPv6," IEEE Internet Computing, vol. 5, pp. 79-81, Jan.-Feb. 2001.

[3] M. Tatipamula and P. Grossetete, and H. Esaki, "IPv6 integration and coexistence strategies for next-generation networks," IEEE Commun. Mag., vol. 42, pp. 88-96, Jan. 2004.

[4] E. B. Fgee, J. D. Kenney, W. J. Phillips, W. Robertson, and S. Sivakumar, "Implementing an IPv6 QoS management scheme using flow label & class of service fields," in Proc. Electrical and Computer Engineering 2004, pp. 1049-1052, May 2004.

[5] IEEE 802.11, " Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: high-speed physical layer extension in the 2.4 GHz band," IEEE Standard, Sep. 1999.

[6] IEEE 802.11, " Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: high-speed physical layer in the 5 GHz band," IEEE Standard, Sep. 1999.

[7] IEEE 802.11, " Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: amendment 4:further higher data rate extension in the 2.4 GHz band, " IEEE Standard, 2003.

[8] S. Ariga, K. Nagahashi, M. Minami, H. Esaki, and J. Murai. Performance evaluation of data transmission using IPSec over IPv6 networks. [Online]. Available: http://www.isoc.org/inet2000/cdproceedings/li/li_1.htm

[9] European Advanced Networking Test Center. Cisco Catalyst 6500 with supervisor 720 – 10 Gigabit Ethernet performance test. [Online]. Available: http://www.eantc.com/press/pressreleases/sep03/EANTC-Summary-Report-Cisco-10GE-Catalyst6500-Supervisor720.pdf

[10] S. Zeadally and I. Raicu, "Evaluating IPv6 on Windows and Solaris," IEEE Internet Computing, vol. 7, pp. 51-57, May-June 2003.

[11] I. Raicu and S. Zeadally, " Impact of IPv6 on end-user applications," in ICT 2003, pp. 973-980, March 2003.

[12] M. Schwartz, "Telecommunication networks: protocols, modeling and analysis," Addision-Wesley, pp. 119-135, Mar. 1987.

[13] E. Modiano, "An adaptive algorithm for optimizing the packet size used in wireless ARQ protocols," Wireless Network, pp. 279-286, May 1999.

[14] M. Degermark, B. Nordgren, and S. Pink, "IP header compression," IETF RFC 2507, Feb. 1999.

[15] M. Engan, S. Casner, and C. Bormann, " IP header compression over PPP," IETF RFC 2509, Feb. 1999.

[16] C. Bormann et al., "RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed," IETF RFC 3095, July, 2001.

[17] M. Degermark, "Requirements for robust IP/UDP/RTP header compression," IETF RFC 3096, July, 2001.

[18] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) specification," IETF RFC 2460, Dec. 1998.

[19] S. Kent and R. Atkinson, "Security architecture for the Internet protocol," IETF RFC 2401, Nov. 1998.

[20] J. H. Yeh, J. C. Chen, and C. C. Lee, "WLAN standards," IEEE Potentials, vol. 22, pp. 16-22, Oct.-Nov. 2003.

[21] J. R. Yee and E. J. Weldon Jr., "Evaluation of the performance of error-corresting codes on a Gilbert channel," IEEE Trans. Commun. Vol. 43, no. 8, pp. 2316-2323, Aug. 1995.

[22] ISO/IEC JTC1/SC29/WG11, "MPEG-4 video verification modelversion 18.0," N3908, Jan. 2001.

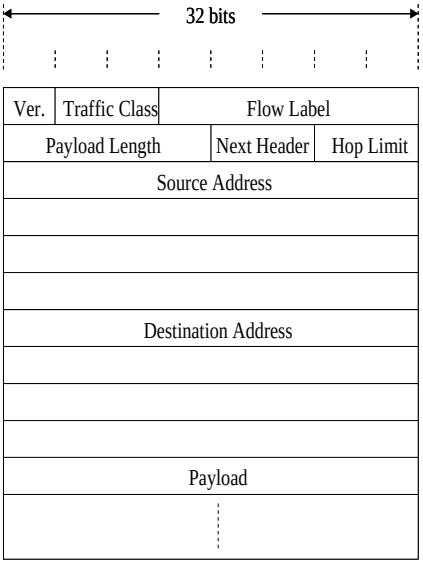

Fig. 1 Structure of IPv6 header

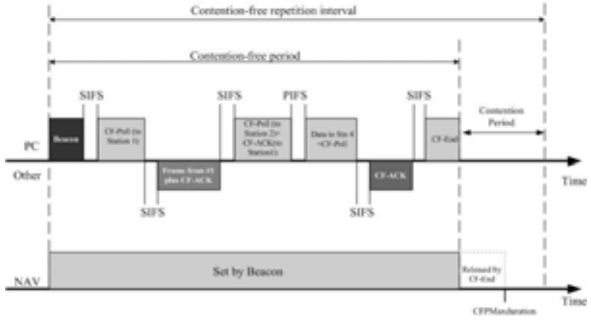

Fig.2 Various data transmission cases in the PCF mode.

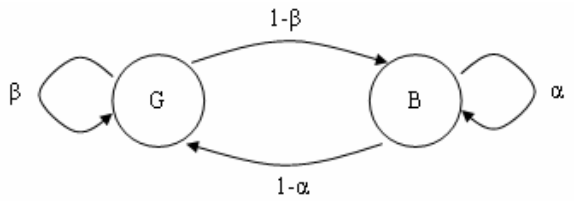

Fig.3 Gilbert model

Table I Parameters of 802.11b/a/g

	802.11 b	802.11 a	802.11 g – OFDM / CCK
Progressing Dealy (τ)	1 μs	<< 1 μs	<< 1 μs
T_{H_PHY}	48 μs	4 μs	4 μs / 48 μs
Preamble Time ($T_{Preamble}$)	144 μs	16 μs	16 μs / 144 μs
Time Slot (T_{slot})	20 μs	9 μs	9 μs / 20 μs
aCW_{min}	31	15	15 / 31
aCW_{max}	1023	1023	1023
Short IFS (T_{SIFS})	10 μs	16 μs	16 μs / 10 μs
PCF IFS (T_{PIFS})	30 μs	25 μs	25 μs / 30 μs
DCF IFS (T_{DIFS})	50 μs	34 μs	34 μs / 50 μs

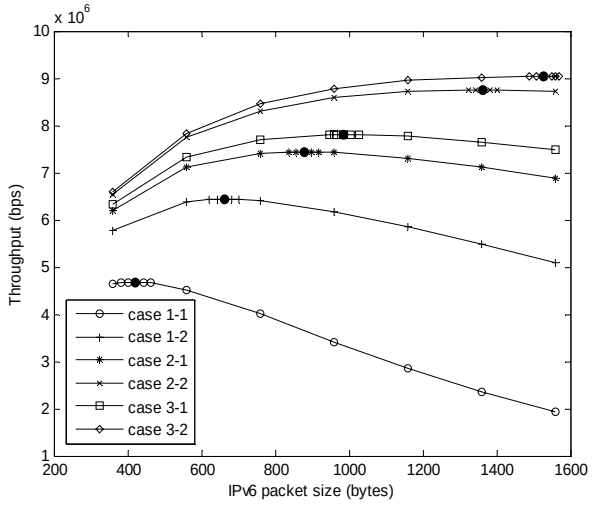

Fig. 4 IPv6 packet sizes versus 802.11g throughputs under different BERs and burst error lengths.

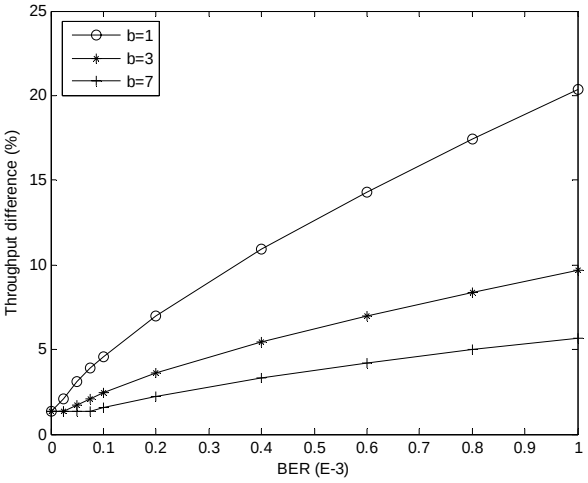

Fig. 6 Throughput comparison of 802.11g under the IPv4 and IPv6 environments.

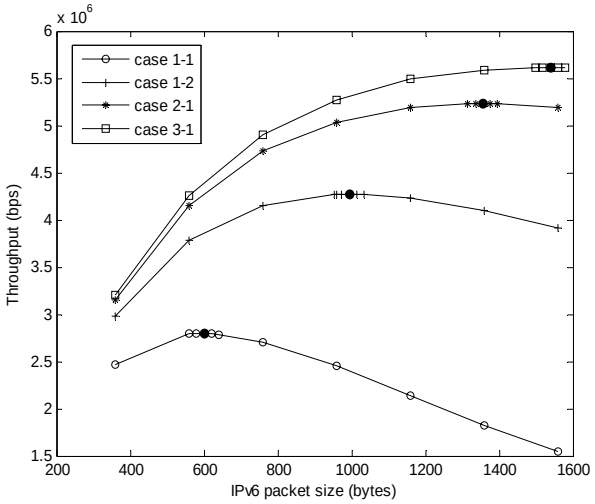

Fig. 5 IPv6 packet sizes versus 802.11b throughputs under different BERs and burst error lengths.

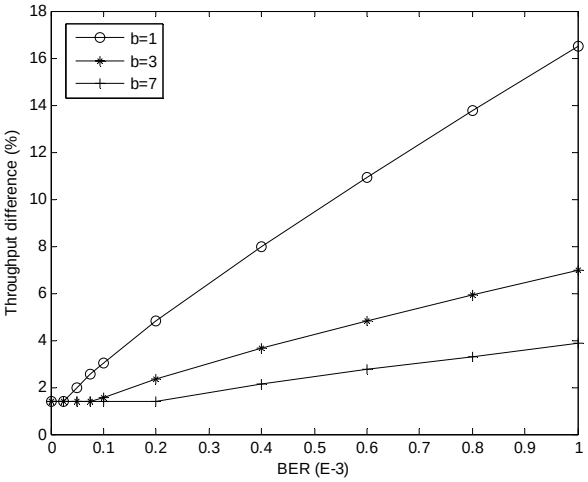

Fig. 7 Throughput comparison of 802.11b under the IPv4 and IPv6 environments

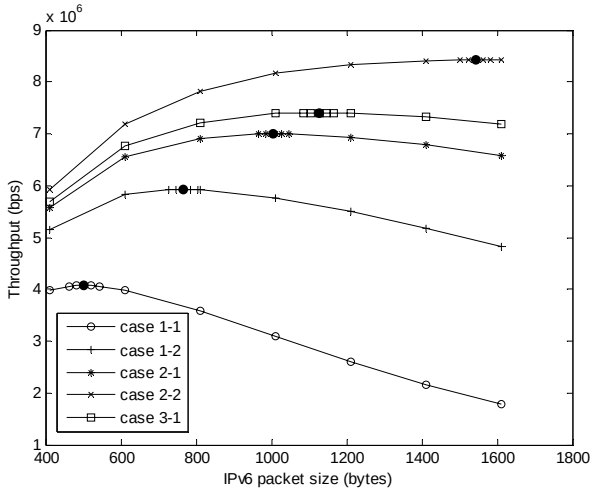

Fig. 8 IPv6 IPSec packet sizes versus 802.11g throughputs under different BERs and burst error lengths.

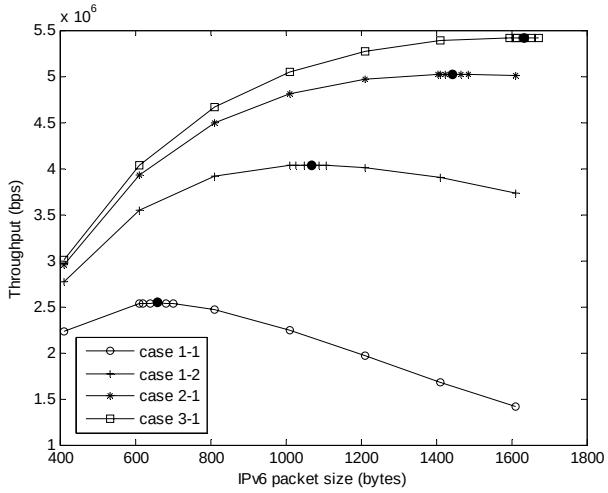

Fig. 9 IPv6 IPSec packet sizes versus 802.11b throughputs under different BERs and burst error lengths.

Table II. Comparison between the available network bandwidth and the actual bandwidth demand of *Foreman*.

Video payload length	1400 bytes			1000 bytes		
Source rate control	Without HOA (IPv4)	Without HOA (IPv6)	With HOA	Without HOA (IPv4)	Without HOA (IPv6)	With HOA
Available network B.W.	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps
Actual B.W. demand	1107.6 Kbps	1125.0 Kbps	1010.4 Kbps	1125.3 Kbps	1148.6 Kbps	1010.5 Kbps
Video payload length	576 bytes			200 bytes		
Source rate control	Without HOA (IPv4)	Without HOA (IPv6)	With HOA	Without HOA (IPv4)	Without HOA (IPv6)	With HOA
Available network B.W.	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps	1000.0 Kbps
Actual B.W. demand	1173.4 Kbps	1212.3 Kbps	1010.9 Kbps	1397.6 Kbps	1505.2 Kbps	1007.3 Kbps