2. CROSS-LAYER CONTROL OF REAL-TIME DATA TRANSMISSION OVER WIRELESS NETWORKS

Real-time multimedia data applications, such as video streaming and video telephony, are regarded as “killer applications” in the emerging wireless networks. Video applications usually involve a large volume of data transmitted in a time sensitive fashion. However, the underlying wireless networks only provide time-varying and limited bandwidth, high data error rate, packet delay and jitter. Extensive research has been done on either video data coding algorithms or wireless network protocols. But the traditional layered network model limits the video transmission over wireless networks because it tries to separate information and functions between different layers. To enable more efficient real-time data transmission over dynamic wireless environments, the applications and underlying wireless networks should cooperate in order to share information and optimize the transmission process dynamically. This chapter reviews the state-of-the-art research efforts on video coding, error control, and rate control algorithms. New cross-layer algorithms are presented which coordinate the algorithms at different layers in order to get better performance than using them separately. The cross-layer rate control algorithm matches the application’s future bandwidth requirement to the available bandwidth in the network so that an optimum data transmission rate can be selected. In the cross-layer error control algorithm, lower layers are responsible for error detection and fast retransmission, while application layer conducts an adaptive error correction algorithm with the help of lower layers.

The chapter is organized as follows. Section 2.1 reviews previous research results. Section 2.2 and 2.3 introduce the cross-layer error control and rate control algorithms, respectively. The theoretical analysis and implementation considerations are also presented in details. Section 2.4 gives the simulation results. Section 2.5 summarizes the chapter and points out some important open problems for future investigation.

2.1 Introduction

In this section, we review previous research results on video transmission over wireless networks, including application-based, network-based and cross-layer approaches.

Application-based Approaches. At application layer, there are two families of standards used to compress the raw video data. The International Telecommunications Union-Telecommunications (ITU-T) H.261 is the first video compression standard gaining widespread acceptance. It was designed for video conferencing. Following H. 261, ITU-T H.263 and its enhanced version H263+ were standardized in 1997 [31], which offered a solution for very low bit-rate (< 64 Kb/s) teleconferencing applications. The Moving Pictures Expert Group (MPEG) series are standardized by International Standard Organization (ISO). MPEG-1 achieves VHS quality digital video and audio at about 1.5 Mb/s. MPEG-2 is developed for digital television at higher bit rate. In contrast, the recently adopted MEPG-4 standard [32] is more robust and efficient in error-prone environments at variable bit rates, which is achieved by inserting resynchronization markers into the bitstream, partitioning macro blocks within each video packet syntactically, using header extension code to optionally repeat important header information, and using reversible variable-length coding such that data can be decoded in both forward and reverse directions. A completely new algorithm, originally referred to as H.26L, is currently being finalized by both ITU and ISO, known as H.264 or MPEG-4 Part 10 Advanced Video Coding (AVC).

Most video applications are characterized as being time sensitive: it will be annoying if the video data does not arrive to the receiver in time. In order to transmit video over error-prone wireless environments, various error resilient video coding algorithms have been proposed [33]. The scalable coding is one of the most important approaches. The scalable or layered video involves a base layer and at least one enhancement layer. The base layer itself is enough to provide usable result, but the enhancement data can further improve the quality. Scalability can be achieved in many forms, including Signal-to-Noise-Ratio (SNR), temporal and spatial scalability. A new scalable coding mechanism, called Fine Granularity Scalability (FGS), was recently proposed to MEPG-4 [34]. An FGS encoder uses bitplane coding to represent the enhancement bitstream. Bitplane coding enables continuous sending rate by truncating the enhancement layer bitstream at anywhere. This advantage of FGS makes it more flexible than other scalable coding algorithms. An important property of scalable coding is that the base layer has the highest priority and must be transported correctly. In contrast, another coding approach, named Multiple Description Coding (MDC) [35], encodes the signal into multiple bitstreams, or descriptions, of roughly equal importance. So any single description can provide acceptable result, while other descriptions complement to each other to produce better quality. Multiple descriptions can be transmitted simultaneously through diverse paths in order to increase the probability of receiving at least one description. MDC achieves these advantages at the expense of coding efficiency.

Rate control and error control are regarded as application-layer QoS techniques which maximize the received video quality in the presence of underlying error-prone networks [33]. Rate control determines the data sending rate based on the estimated available bandwidth. A lot of so called "TCP-friendly" rate control approaches have been proposed for best-effort internet in order to avoid network congestion [6]. Error control is employed to reduce the effect of transmission error on applications. Two basic approaches are Forward Error Correction (FEC) and Automatic Repeat Request (ARQ). FEC adds parity data to the transmitted packets and this redundant information is used by the receiver to detect and correct errors. FEC maintains constant throughput and has bounded time delay. ARQ is based on packet retransmissions when errors are detected by the receiver. ARQ is simple but the delay is variable. Many alternatives to FEC and ARQ have been introduced in [36]. One of the most well-known error coding techniques is Reed-Solomon coding [37] which deals with burst errors. The field of R-S coding is of the form GF(2^M), where M is any positive integer. Each bit block of 2^M bits is called a symbol. If the original packet length is K symbols, then after adding redundant parity data, the codeword will be of length $N > K$. The original packet can be completely recovered when there are no more than $(N – K) / 2$ error symbols during transmission. In order to protect source data with different importance, joint source/channel coding has been proposed [30]. For example, different frames in MPEG-4 coding or different layers in scalable source coding can combine with unequal length of parity data. Error concealment [10] is a technique employed by receivers to minimize the effect of packet errors on the quality of video.

Network-based approaches. In addition to the research on video applications, a large body of research has been conducted on improving underlying networks for the benefits of upper applications. We will review network-based approaches from right below application layer down to physical layer, according to the OSI reference model.

Between application layer and transport layer, there are several standardized protocols designed for supporting real-time applications, such as real-time transport protocol (RTP), real-time control protocol (RTCP), real-time streaming protocol (RTSP) and session initiation protocol (SIP) [33]. RTP provides extra information to application layer in the form of sequence numbers, time-stamping, payload type, and delivery monitoring. But RTP itself does not ensure timely delivery or other QoS guarantees. RTCP is a control protocol for monitoring RTP packet delivery. RTSP and SIP are designed to initiate and direct delivery of real-time data.

At transport layer, TCP provides reliable transmission of data by flow control, congestion control and retransmission. However, for most real-time communications, applications can tolerate data errors to some extent, but they have strict time constraint. So another simpler transport protocol, UDP, is widely used for real-time data transmission. UDP only uses cyclic redundancy check (CRC), or checksum, to verify the integrity of received packet. Since UDP does not perform any error correction, it may sacrifice the whole packet only for some minor data errors, which can yield unpredictable degradation and poor application quality. In order to solve this problem, a modified version, called UDP Lite, is introduced [39]. UDP Lite allows partial checksum on packet data by enabling application layer to specify how many bytes of the packet are sensitive and must be protected by checksum. If bit errors occur in the sensitive region, the receiver drops the packet; otherwise it is passed up to the application layer. This approach allows the application to receive partially corrupted packets which may still generate acceptable video quality.

Most differences between wireless and wired networks exist below transport layers. Wireless networks involve different kinds of radio access networks, such as mobile cellular network and WLAN. Under the umbrella of IMT-2000, 3G mobile network standards provide high date rate up to 384 kbps for mobile users and 2 Mbps for users at pedestrian speed. One of the most promising 3G networks is Wideband CDMA (WCDMA), also known as Universal Mobile Telecommunications System (UMTS). UMTS is standardized by 3GPP [40]. A UMTS network consists of Universal Terrestrial Radio Access Network (UTRAN) and core networks (CN). An IP packet coming from an internet host is first directed to a 3G Gateway GPRS Support Node (3G-GGSN) through Gi interface, then tunneled from 3G-GGSN to UTRAN via 3G Serving GPRS Support Node (3G-SGSN) by GPRS Tunneling Protocol for data transfer (GTP-U). We are most interested in data transmission over the wireless link, which is the air interface (Uu) between UTRAN and mobile stations (called UE in UMTS), because the data transmission rate at wireless link is usually the bottleneck of throughput. Radio Link Control (RLC) [41], located at the upper half of data link layer, segments packets into radio blocks and provides three data transfer modes to upper layers: Transparent Mode without any extra information; Unacknowledged Mode which can only detect erroneous data; Acknowledged Mode which provides more reliable data delivery by limited number of retransmissions. Media Access Control (MAC) is responsible for mapping logical channels into transport channels provided by physical layer (PHY), as well as reporting of measurements to the Radio Resource Control (RRC). RRC manages and controls the use of resources and therefore has interactions with RLC, MAC and PHY. In addition to wide area wireless networks, WLANs have been rapidly accepted in enterprise environments, mainly due to the standardization by IEEE 802.11 work group [42] and the low cost of deploying a WLAN. The IEEE 802.11a and the newly approved 802.11g can provide high data rate up to 54 Mbps. IEEE 802.11 standards only define the physical layer and MAC layer in order to seamlessly connect WLAN to existing wired LANs. For the channel access control, in contrast to the centralized control in cellular networks, IEEE 802.11 employs carrier sense multiple access with collision avoidance (CSMA/CA) in a highly distributed fashion. The mandatory function implemented in 802.11 stations is distributed coordination function (DCF), by which mobile stations contend for the radio channel and there is only one user at one time. But the DCF does not fit well to the time-sensitive applications. Instead, another optional point coordination function (PCF) should work better because there is a point coordinator (PC) which polls every user in the contention free period (CFP). The contention period (CP) controlled by DCF and the CFP controlled by PCF alternate to accommodate both functions. At the physical layer, IEEE 802.11 provides different transmission rates. For example, 802.11a provides eight data rates from 6 Mbps to 54 Mbps by changing channel coding parameters.

Research activities on video transmission over wireless networks can be found in [43] - [59]. Liu [43] discussed rate control of video source coding for wireless networks. Wu [44] proposed a general adaptive architecture for video transport over wireless networks. Vass [45] proposed a novel joint source/channel coding for wireless links, based on the concept of application-level framing. Wang [46] compared MDC with scalable coding in wireless networks when multiple paths are available. Majumdar [47] investigated unicast and multicast video streaming over WLAN where the wireless network is modeled as a packet erasure model at network layer. Miu [48] presented experimental results to show the advantage of transmitting H.264/MPEG-10 AVC encoded video over multiple access points in IEEE 802.11b WLAN. The above approaches are used at application layer and the underlying wireless network is considered as a high packet loss rate environment. Other research is focused on improving the performance of wireless networks. Fitzek [49] proposed a prefetching protocol for continuous media streaming between a base station and mobile stations in a cell. The protocol used transmission rate adaptation techniques to dynamically allocate transmission capacity for different streams. Singh [50] evaluated the performance of using UDP Lite and transparent mode RLC in GSM. Wang [51] developed a new MAC protocol for FDD WCDMA in 3G cellular networks. A scheduling scheme, assisted by minimum power allocation and effective connection admission control, was used to fairly queue multimedia traffic with different QoS constraints. Zhang [52] considered scalable video transmission over pre-coded Orthogonal Frequency Division Multiplexing (OFDM) system, enhanced by adaptive vector channel allocation. Zou [53] proposed a MPEG-4 frame drop policy in order to save bandwidth of IEEE 802.11 WLAN in the DCF mode.

Cross-layer approaches. According to the above review, most current research activities are focusing on solving part of the problem: there have been a huge number of results based on applications and networks separately, but there is still not much research on the interaction between different layers in order to take full advantage of each other. Girod [54] introduced several advances in channel-adaptive video streaming, which involved several system components jointly. But their integration to wireless networks was not investigated. Zhang [55] integrated their work at different layers into a cross-layer framework supporting multimedia delivery over wireless internet. However, they did not address the cooperation of techniques at different layers. Zheng [56] introduced an improved UDP protocol. The protocol captures the frame error information from link layer and uses it for application layer packet error correction. Shan [57] proposed a cross-layer error control scheme in which the application layer implements FEC and requests link layer to retransmit lost packet when necessary. Ding [58] proposed a cross-layer error control algorithm which can dynamically adjust FEC at application layer according to the link layer retransmission information. Krishnamachari [59] investigated cross-layer data protection for IEEE 802.11a WLAN in the PCF mode. The application layer scalable coding and FEC were adopted, along with MAC layer retransmission and adaptive packetization. The above results investigated cross-layer error control methods in order to deal with high error rate in wireless networks. How to use limited and variant bandwidth in wireless networks is another challenging issue that should be addressed. In the following sections, we will introduce the cross-layer error control and rate control algorithms for real-time data transmission over mobile wireless networks.

2.2 Cross-Layer Rate Control

At data link layer, the link-adaptation techniques in wireless networks [60, 61] can be employed to adjust transmission bandwidth. At application layer, the data rates of real-time applications can also be adjusted according to different QoS requirements (e.g., layered application). In this section, we introduce a cross-layer transmission rate control algorithm involving both the application layer and the radio access network. The future data rate requirement of the application is used to determine the network transmission rate.

2.2.1 Rate control algorithm

In the data link layer of wireless networks, transmitting rate is determined according to the size of the buffered data at the transmitting site. When the buffer size is too large, a higher transmission rate is needed. When the buffer size is too small, the transmission rate will be reduced. Compared to the cross-layer rate control algorithm, the scheme on monitoring buffer size is lack of reality since it only reflects the past bandwidth requirements. The comparison is illustrated in Figure 2.1. When the application data rate increases during the time interval (*t*, *t* + *T*), the lower layer buffer is rapidly overflowed at point A. The bandwidth adjustment is triggered so that, at point A, the transmitting rate is high enough for the application. However, the bandwidth needed for application continues to increase after point A, and another bandwidth adjustment might be needed at point B. Using cross-layer rate control, the bandwidth requirement for time (*t*, *t* + *T*) can be obtained from application layer at point C, and the accurate transmitting bandwidth can be assigned in one step.

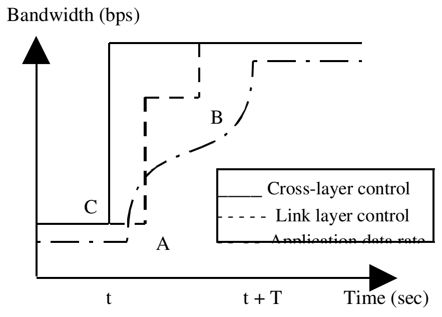


Fig. 2.1 Illustration of using cross-layer

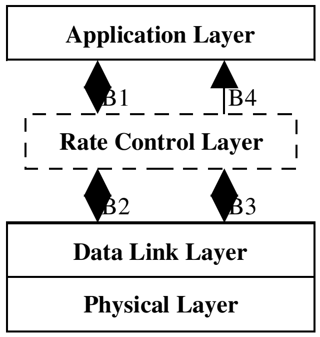


Fig. 2.2 Protocol architecture for cross-

rate control layer rate control

The cross-layer rate control algorithm is suitable for layered applications. To address this, we assume that a video application has *n* layers of data streams, and each extra layer enhances the quality of previous layers but requires more bandwidth. We also assume that the available transmitting bandwidth set in the network is Φ. The protocol architecture is shown in Figure 2.2. Note that the rate control layer is a virtual layer, which can actually be implemented at any appropriate layer for a particular wireless network. Control steps in Figure 2.2 are explained as follows:

B1. Request bandwidth requirement for the coming time interval from the application layer:

$$\boldsymbol{BW_H}(t,\ t+T) = [BW_{H1}, BW_{H2}, \ldots, BW_{Hn}].$$

BW_{Hi} is the bandwidth requirement for the application with *i* layers.

B2. Request current network bandwidth from lower network layers: $BW_L \in \Phi$.

B3. Call procedure *BWSchedule*($\boldsymbol{BW_H}$, BW_L) with return values (k, BW_L^{new}).

B4. Inform application layer of the maximal layer, *k*, that can be sent, as well as the current network bandwidth BW_L^{new}.

```
procedure BWSchedule(BW_H, BW_L)
    if $\int_t^{t+T} BW_{H_n}(\tau)d\tau - BW_L \cdot T < Buf^-$ then
        $BW_L^{new} = \min\{BW \in \Phi \mid BW \le BW_L,$ and $BW \ge \frac{1}{T}(\int_t^{t+T} BW_{H_n}(\tau)d\tau - Buf^+)\}$
        request $BW_L^{new}$ from lower layers
        return (n, $BW_L^{new}$)
    endif
    if $\int_t^{t+T} BW_{H_n}(\tau)d\tau - BW_L \cdot T > Buf^+$ then
```

```
        if $BW_L^{new} = \min\{BW \in \Phi \mid BW \geq \frac{1}{T}(\int_t^{t+T} BW_{H_n}(\tau)d\tau - Buf^+)\}$ exists then
            request $BW_L^{new}$ from lower layers
            return (n, $BW_L^{new}$ )
        else
            $BW_L^{new} = \max \Phi$
            request $BW_L^{new}$ from lower layers
            if $k = \max\{i \in [1,\ldots,n] \mid \int_t^{t+T} BW_{H_i}(\tau)d\tau \leq T \cdot BW_L^{new} + Buf^+)\}$ exists then
                return (k, $BW_L^{new}$ )
            else
                return (1, $BW_L^{new}$ )
            endif
        endif
    else
        return (n, $BW_L$)
    endif
endprocedure
```

Fig. 2.3 Procedure *BWSchedule*

In this rate control algorithm, the available bandwidth BW_L and the future bandwidth requirement $\boldsymbol{BW_H}$ $(t,\ t + T)$ are checked periodically. If the future bandwidth requirement reaches the upper limit Buf^+ of the buffer, a larger available bandwidth BW_L^{new} that can satisfy the requirement is found. If the maximum bandwidth in Φ cannot satisfy the requirement for all of *n* layers of the application, a maximal number *k* is found so that *k* layers of the application can be accommodated using the available maximum bandwidth. Transmitting bandwidth will be reduced when the future application bandwidth requirement is so small that the negative buffer limit, Buf^-, is reached. The procedure *BWSchedule* is given in Figure 2.3. It runs at every time $t = jT$, where T is the time period and j is a non-negative integer.

2.2.2 Analysis

According to the above cross-layer rate control algorithm, since it always chooses the minimal transmission bandwidth that can satisfy the future application requirements, it will neither waste unnecessary bandwidth, nor fail to provide enough bandwidth to upper applications as far as there is some available bandwidth that can satisfy the requirement. This proves the efficiency of the proposed rate control algorithm. In addition, if the elements in Φ have been pre-ordered, the complexity of the algorithm is O(log(max{n, m})) which is computationally tractable even for large n and m.

One practical issue with this algorithm is that, when it requests a bandwidth $BW_L^{new} \in \Phi$ from lower layers, it may not guarantee to actually obtain this bandwidth at any time. For instance, other applications might have used up all the channels providing the requested transmission capacity, the interference from other users might degrades the quality of requested channel, it might take too much time to communicate with radio access networks to set up the requested radio link, and so on. To theoretically analyze this issue, we assume that there are some other elements, in the pre-ordered set Φ, that are between current bandwidth BW_L and the requested bandwidth BW_L^{new}. At every intermediate bandwidth, it has probability $0 < p < 1$ to reach its neighbor bandwidth closer to BW_L^{new} and probability $1 - p$ to remain the current bandwidth. Under these assumptions, we have the following statement:

The requested bandwidth BW_L^{new} *can finally be granted with probability 1, but the cost is 1/p times of the case when every bandwidth request can be immediately granted.*

This statement can be theoretically proven by discrete markov chain and Wald's equation. We omit the details due to space limit.

2.2.3 Implementation considerations

The cross-layer algorithms involve at least two layers. This fact may increase the difficulty of implementing them. We here discuss the implementation of the proposed cross-layer algorithm in two most popular wireless networks: 3G UMTS and IEEE

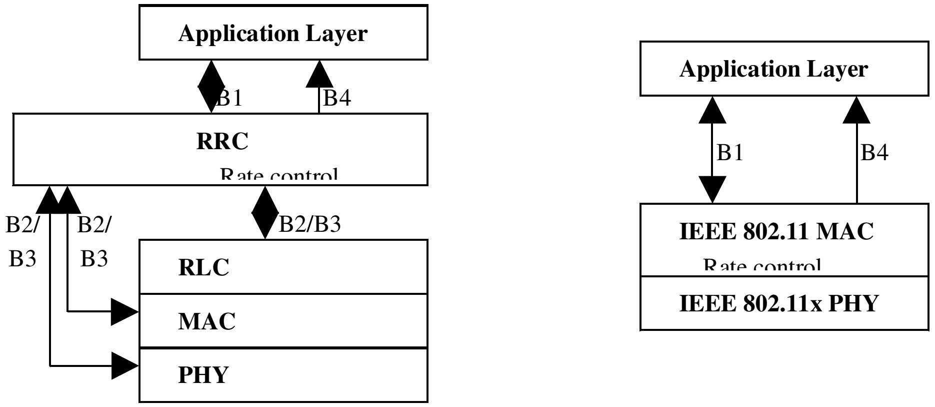
802.11 WLAN. Figure 2.4 shows the recommended implementation of the proposed cross-layer rate control algorithm in both wireless networks. Since RRC has the control interfaces to RLC, MAC and PHY, the proposed rate control algorithm can be implemented in RRC. Along with UTRAN, RRC can control the radio channel configuration in different lower layers in order to satisfy different bandwidth requirements [60]. Examples of the channel configuration for increasing bandwidths are as follows:

i) For time-bounded applications, if a common channel is initially used, it can be reconfigured at physical layer as a dedicated channel.

ii) An appropriate transport format (TF) can be chosen from the defined transport format combination (TFC) for the dedicated channel.

iii) The radio link can be further reconfigured at MAC layer.

The detailed radio channel configuration in RRC and UTRAN is out of the scope of this chapter.



(a) In UMTS (control plane) (b) In IEEE 802.11 WLANs

Fig. 2.4 Implementation of cross-layer rate control

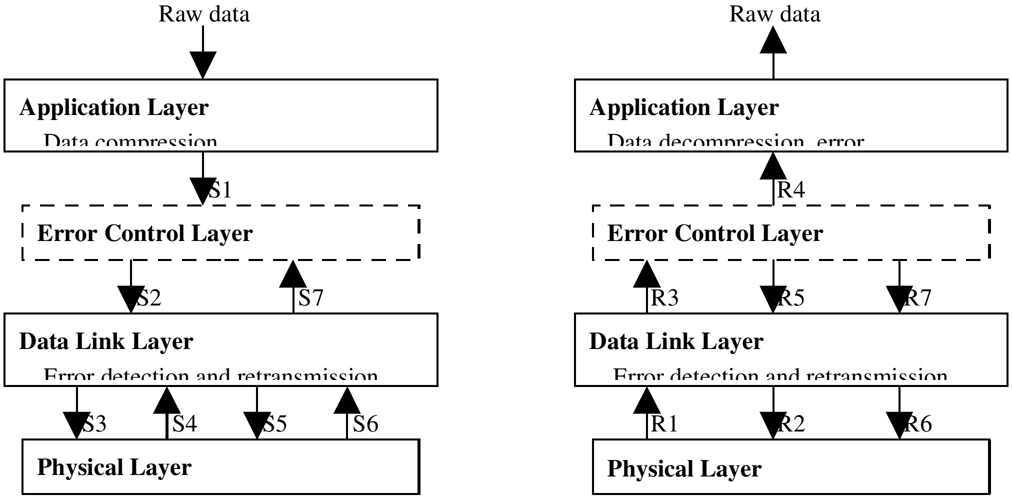
In IEEE 802.11 WLANs, the cross-layer rate control algorithm can be implemented at MAC layer. The cross-layer communications include getting future bandwidth requirement from the application layer (by step B1) and sending rate control results to the application layer (by step B4). The physical layer of IEEE 802.11 WLAN provides multiple bandwidth choices. Through link adaptation techniques [34], different transmission rates can be chosen according to the application requirements. A great deal of research is undergoing to further improve the QoS capacity in IEEE 802.11 MAC layer. Although the PCF mode is more suitable for video applications than DCF, it does not work in the contention period. A new standard, IEEE 802.11e [62], is currently under development. In IEEE 802.11e, DCF is replaced by Enhanced Distributed Channel Access (EDCA) and PCF is replaced by hybrid coordination function (HCF). EDCA provides QoS support by assigning shorter inter-frame space to higher priority traffic categories, while the HCF enables poll-based data transmission even in contention period.

2.3 Cross-Layer Error Control

Due to channel fading, user mobility and interference from other users, wireless links are less reliable than wired connections. Data link layer and physical layer in wireless networks provide some error control mechanisms, such as error detection and packet retransmission. Yet the lower layers do not know to which extent their error control mechanisms should be used, which is actually better known by the upper layer applications. For example, when using retransmission for error correction in a video application, the lower layers only know whether the data is received correctly, based on which the data will be re-sent or not. Whether the delay is within the time constraint is judged by the application layer. Error control techniques can also be embedded into applications. However, application-layer retransmission is usually less efficient than link layer retransmission [55], and FEC at application layer can not adapt to network conditions to reduce the redundant error control information. In this section, a novel cross-layer error control algorithm is described. We will go through the algorithm in details, followed by the performance analysis, and implementation considerations.

2.3.1 Error control algorithm

The algorithm with inter-layer signaling is illustrated in the protocol stack in Figure 2.5.



(a) Sender (b) Receiver

Fig. 2.5 Protocol architecture for cross-layer error control

The details for the steps of the algorithm at a sender are as follows:

S1. The compressed real-time data is first processed by error control layer. Data is segmented into packets of K bytes. For each packet, R bytes of parity data are added by R-S coding in the field GF(2^8), and a small header (including sequence number n and parity data length R) of H bytes is also added. The data is then saved into a buffer of $N \cdot (K + R + H)$ bytes row by row and is read column by column. This is for data interleaving. The packet size of the output data is of N bytes. Parameter R is adjusted by

$$R = \max(R - R_{step}, R_{min}), \qquad (2.1)$$

where R_{min} and R_{step} are the minimal value and decreasing/increasing step of R, respectively.

S2. At data link layer, packet n is fragmented into small radio frames with sequence numbers $n_1, n_2, \ldots$, and n_m. CRC is added in each radio frames for error detection.

S3. A radio frame n_k is sent and a timer is set.

S4. If a non-acknowledgement (NACK), including a type LOW and a frame sequence number n_k, is received from the receiver, or the timer is timeout,

S5. Then frame n_k is retransmitted. If the sender keeps getting NACKs for the same frame after retx_{max} times of retransmissions, this frame will be discarded and its timer is cleared.

S6. If a NACK, including a type HIGH and a packet sequence number n, is received, it is forwarded to error control layer.

S7. At error control layer, upon receiving a type HIGH NACK with a packet sequence number n, R is updated by

$$R = \min(R_0 + 2^k R_{step}, R_{max}), \tag{2.2}$$

where R_0 and R_{max} are the initial value and the maximal value of the length R of the parity data, respectively. k is increased by 1 if the packet sequence number in NACK immediately follows the packet sequence number in last NACK. Otherwise k is set to 0.

The details for the steps of the algorithm at a receiver are as follows:

R1. When a frame n_k is received, errors are detected by CRC at the link layer. If any error is detected or the received frame is out of sequence,

R2. Then a NACK, including a type LOW and the frame sequence number n_k, is sent back to the sender. If no error is detected, all frames belonging to packet n are assembled. If the packet n is completed, it is sent up to error control layer.

R3. At error control layer, the received packets are interleaved in order to get the original application packets. R-S decoding algorithm is run to correct errors if there is any. On receiving a duplicated packet, the old packet is replaced with the new one.

R4. The packet is buffered until it is retrieved by the application layer. The application layer uses its own error concealment techniques if an error can not be recovered by the lower layers.

R5. At error control layer, if a packet cannot be corrected, a NACK including the type HIGH and the sequence number n of the erroneous packet is sent down to data link layer.

R6. At data link layer, upon receiving a NACK from error control layer, the NACK is sent to the sender.

R7. If a packet n is still not available when it is requested by the upper applications, the error control layer sends a request to the data link layer for this packet. The data link layer will immediately send the packet n to upper layers and go to step 3, even if the requested packet is not completed yet.

There are several remarks on the above algorithm.

First, for the adaptive error control algorithm, the larger the length R of parity data, the more errors can be corrected. On the other hand, this redundancy reduces the bandwidth efficiency. An appropriate R has to be chosen. In the proposed adaptive algorithm, R is dynamically adjusted based on the network information. When there is no HIGH NACK received, sender periodically decreases R from its initial value R_0 by a small step R_{step}. When there is a type HIGH NACK received, which indicates that a higher level error protection is requested, the sender sets R back to R_0. If another HIGH NACK is received immediately following the last HIGH NACK, R is exponentially increased by (2.2). This is used to enhance the error protection capacity in response to the error burst in wireless channels. The initial value of the parity data length R_0 is calculated by

$$R_0 = a\, R_0 + 2\,(1 - a)\, b\, (K + R)\, C_{error}/C_{total}, \quad (2.3)$$

where C_{error} and C_{total} are the number of received HIGH NACKs and the total number of sent packets, respectively, and C_{error}/C_{total} is the estimation of packet loss rate. b deals with the variance of the estimation and $b > 1$. a is used to smooth the estimation and $0 < a < 1$. This algorithm works when packet loss rates do not vary very fast.

Second, since each packet can be protected by parity data using different values of R, the proposed algorithm can provide multiple levels of error protection. In an application when data has different importance, the most important part can be assigned a larger R_0 by setting a larger b in (2.3), while less important part can be protected by a smaller R_0.

Third, in Figure 2.5, for the sake of illustration and generality, an error control layer is shown between the application layer and the data link layer. It can be actually implemented in application layer, data link layer, or any appropriate layers in between. This virtual error control layer serves as a controller which coordinates the error control techniques in different layers. Its operations include:

i) When the upper layer FEC cannot recover a packet, it requests lower layers to send a type HIGH NACK to the sender in order to increase the parity data length.

ii) The existing lower layer protocols do not send a packet to the upper layers until every fragment of the packet has been correctly received. This may incur unnecessary packet delay, which is not deserved by real-time services. In the proposed error control algorithm, the virtual control layer can request the lower layer to send an incomplete packet when the upper application requires. The upper layer will try to recover the corrupted packet by its own FEC. In this way, the error control techniques in both the upper layer and the lower layer are fully utilized.

iii) Retransmissions in the link layer can be executed up to $\mathrm{retx}_{\max}$ times or until the packet is requested by upper layers. Since retransmission can be stopped by the upper applications, it does not incur any extra packet delay. However, it increases the probability of obtaining a correct packet before it is requested.

Fourth, Figure 2.5 does not show the intermediate network layers between the application layer and the data link layer. If IP is used for the network layer, IP header is protected by CRC. Packets with corrupted IP header will be discarded. At the transport layer, UDP Lite should be employed in order to pass data payload up as far as there is no error in the header.

2.3.2 Analysis

In order to analyze the above error control algorithm, we adopt the well-known two-state Gilbert Elliot model to represent the variation of wireless channel. The wireless channel can stay at two states: "good" state and "bad" state. The bit error rate (BER) at "good" state, defined as e_g, is far less than the BER at "bad" state, defined as e_b. The state transition matrix is

$$\begin{bmatrix} \mu_{gg} & 1-\mu_{gg} \\ 1-\mu_{bb} & \mu_{bb} \end{bmatrix}, \tag{2.4}$$

where μ_{gg} and μ_{bb} are the transition probabilities of staying in "good" and "bad" states, respectively. The steady-state BER can be calculated by:

$$p_b = \frac{1-\mu_{gg}}{2-\mu_{gg}-\mu_{bb}} e_g + \frac{1-\mu_{bb}}{2-\mu_{gg}-\mu_{bb}} e_b \,. \tag{2.5}$$

The radio frame error probability at data link layer is

$$p_L = 1-(1-p_b)^{M0+H0} \approx (M_0+H_0)p_b \,, \tag{2.6}$$

where M_0 and H_0 are the length of data payload and header of the radio frames, respectively. In order to reflect the effect of retransmission on the system, we define a transmission efficiency parameter as the ratio of transmission time without loss and the transmission time with loss-and-retransmission:

$$\gamma_L = \frac{(M_0+H_0)}{\sum_{n=1}^{retx\max-1} (nM_0+(2n-1)H_0)P_{n0} + (retx_{\max}M_0+(2retx_{\max}-1)H_0)(1-\sum_{k=1}^{retx\max-1} p_{n0})}$$

, (2.7)

where $P_{n0} = p_L^{n-1}(1-p_L)$ represents the probability of n – 1 radio frames getting errors before a successful (re)transmission. We assume that the transmission rate is constant during retransmissions and the size of NACK packets is dominated by the header length.

At error control layer, due to interleaving, every lower layer frame loss corresponds to one byte error in the application packet. An application packet loss occurs when there is an error in the header of the packet, or there are more than *R*/2 corrupted bytes in the payload. The probability of a packet loss can then be calculated as:

$$p_H = (1-(1-p_L)^H)\frac{H}{M+H} + \sum_{i=R/2+1}^{M}\binom{M}{i} p_L^i(1-p_L)^{M-i}\frac{M}{M+H}, \tag{2.8}$$

where $M = K + R$ is the length of data payload. A discrete Markov Chain can be used to model the state of R. Specifically, we let state 0 represent initial value R_0 and state i represent $R = R_0 + i^*R_{step}$, where $i \in \Gamma = [N_L, 2^{NH}]$. N_L is the negative lowest bound and 2^{NH} is the upper bound. Letting $p_H(i)$ denote p_H when $R = R_0 + i^*R_{step}$, the non-zero elements in the transition matrix $\{p_{i,j}\}_{i,j\in\Gamma}$ for the Markov Chain is:

$$p_{NL,NL} = 1 - p_H(N_L), \text{ and } p_{NL,0} = p_H(N_L),$$

$$p_{i,i-1} = 1 - p_H(i), \text{ and } p_{i,0} = p_H(i), \quad \text{for } N_L < i < 0,$$

$$p_{0,-1} = 1 - p_H(0), \text{ and } p_{0,1} = p_H(0),$$

$$p_{i,i-1} = 1 - p_H(i), \text{ and } p_{i,2^h} = p_H(i), \quad \text{for } 2^{h-1} \le i < 2^h, \ h = 1,2,\ldots,N_H,$$

$$p_{2^{NH},2^{NH}-1} = 1 - p_H(2^{NH}), \text{ and } p_{2^{NH},2^{NH}} = p_H(2^{NH}).$$

The state transition graph for N_L = -3 and N_H = 3 is depicted in Figure 2.6.

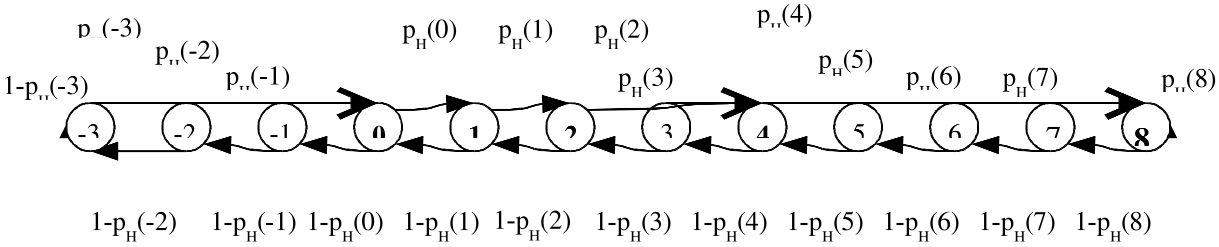


Fig. 2.6 Transition graph of Markov Chain model (N_L = -3 and N_H = 3)

The stationary distribution $\boldsymbol{\pi}$ can be found as

$$\pi(i) = x(i)\pi(0), \quad i \in \Gamma,$$

where

$$\pi(0) = 1\Big/\sum_{i\in\Gamma} x(i),$$

$$x(N_L) = \frac{1}{p_H(N_L)}\prod_{j=NL+1}^{0}(1-p_H(j)),$$

$$x(i) = \prod_{j=i+1}^{0}(1-p_H(j)), \quad \text{for } N_L < i < 0,$$

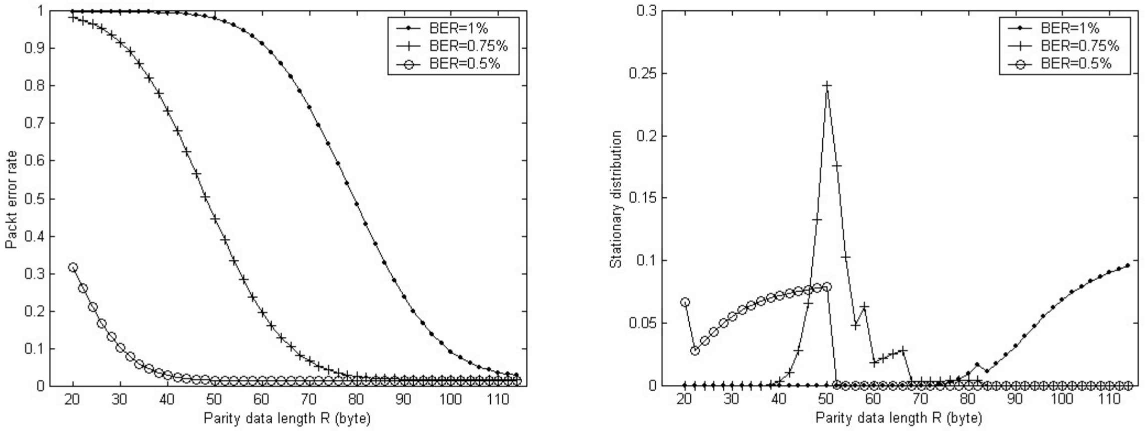
$$x(0) = 1, \quad x(1) = \frac{p_H(0)}{(1-p_H(1))}, \quad x(2) = \frac{p_H(0)p_H(1)}{(1-p_H(1))(1-p_H(2))},$$

$$x(2^{h+1}) = \frac{1}{\prod_{j=2^{h}+1}^{2^{h+1}}(1-p_H(j))}((1-\sum_{k=2^{h-1}+1}^{2^{h}-1} p_H(k)\prod_{j=k+1}^{2^{h}}(1-p_H(j)))x(2^h)-p_H(2^h-1)x(2^h-1))$$

, for $h = 1,\ldots,(N_H - 1)$

and $x(i) = \prod_{j=i+1}^{2^h}(1-p_H(j))x(2^h)$, for $2^{h-1} < i < 2^h$, $h = 2,\ldots,N_H$.

Figure 2.7 illustrates the packet loss rate and stationary distribution when $N_L = -15, N_H = 5$ and *R₀* = 50 bytes. The distribution of *R* is dynamically changing according to the network conditions. For example, when BER is as high as 1%, the parity data length *R* is more likely to have values larger that R_0. But when BER is equal to 0.5%, *R* tends to be smaller than R_0. This demonstrates the efficiency of the proposed adaptation algorithm.



(a) Packet error rate (b) Stationary distribution of *R*

Fig. 2.7 Packet error rate and distribution of parity data length *R*

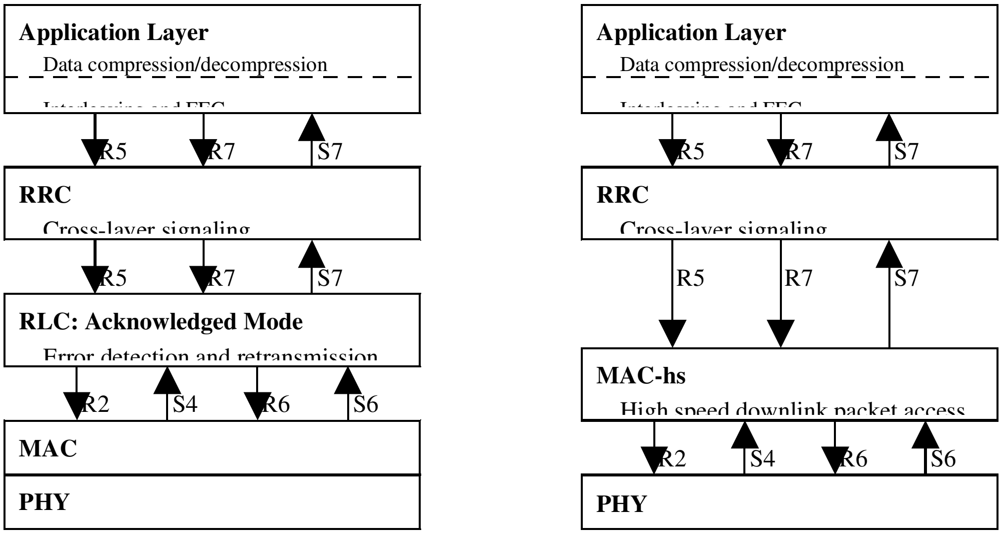
2.3.3 Implementation considerations

In UMTS, two implementations are recommended, which use RLC and high speed downlink packet access (HSDPA) [63] for link layer error control, respectively. The modified protocol architectures are shown in Figure 2.8. Only steps in control plane are shown. Other steps belong to data plane and can be treated as regular data transmission in UMTS. The major procedures of implementing cross-layer error control algorithm using RLC include follows:

i) The upper layer FEC and interleaving can be integrated into the application layer where developers have more freedom.

ii) The RLC layer of UMTS can be modified to enable sending NACK frames so that it can handle data link layer ARQ for the cross-layer control algorithm. The related control steps for a sender and a receiver are steps S4, S6, R2 and R6 in Figure 2.5.

iii) Radio resource control (RRC) is located at Layer 3 (network layer) of UTRAN and has the control interfaces to RLC, MAC and PHY. RRC can also provide services to upper layers. Therefore, RRC can function as relaying cross-layer control signals. The related controls steps are R5, R7, and S7.



(a) Error control using RLC (b) Error control using HSDPA

Fig. 2.8 Implementation of cross-layer error control in UMTS (control plane)

HSDPA is designed to enhance downlink data transmission capacity in UMTS. A new transport channel between MAC and PHY, named shared HSDPA data channel (HS-DSCH), is introduced. HS-DSCH is only used for high speed downlink data transmission from a base station to UE. It is controlled by a new entity in MAC, called MAC-hs, which employs hybrid ARQ (HARQ) to make retransmission decision. Since MAC-hs is closer to physical layer than RLC is, applying error control in MAC-hs is expected to be more efficient than that in RLC. The corresponding control plane protocol stack is shown in Figure 2.8b.

In IEEE 802.11 WLANs, all the radio access work is conducted in MAC and PHY layers in order to make the underlying networks transparent to the upper applications. This makes it difficult to enable cross-layer information exchange. To minimize cross-layer signaling, the protocol architectures in Figure 2.9 are recommended. Following aspects are discussed: the type HIGH NACK can be an application packet so that step R5 and S7 do not incur any inter-layer communications. HIGH NACK is used to notify the sender of an FEC failure. Yet it does not request any retransmissions at application layer. Since the ACK-based retransmission is mandatory in IEEE 802.11 MAC layer, the MAC layer can be modified by adding the type LOW NACK. This can be implemented by adding an extra field of frame sequence number in the ACK MAC frame. This may also be implemented as a new control MAC frame using one of the reserved subtype numbers between 0001 to 1001 in the field of "frame control" [14].

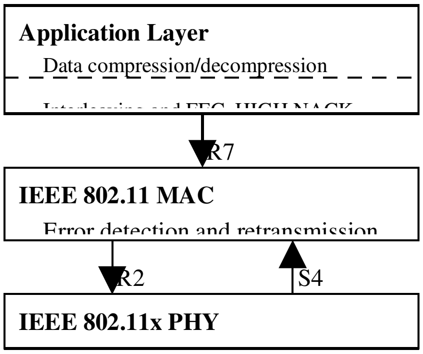

Application Layer
Data compression/decompression
Interleaving and FEC, HIGH NACK
R7
IEEE 802.11 MAC
Error detection and retransmission
R2
S4
IEEE 802.11x PHY

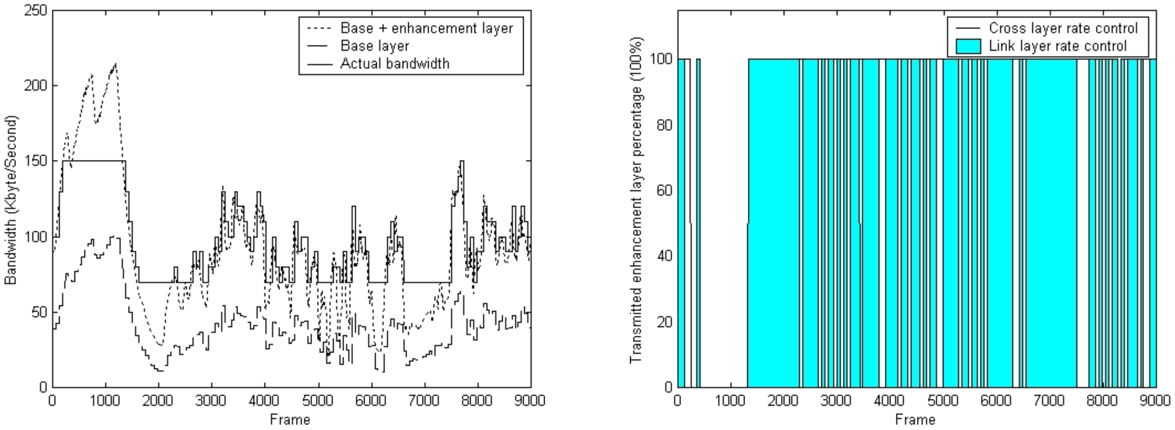
Fig. 2.9 Implementation of cross-layer error control in IEEE 802.11 WLANs

2.4 Results

The proposed algorithm is extensively tested by simulations on video transmission over wireless links. The video traces are taken from ten movies, one-hour each [64]. They are generated specifically for the research on video transmission over wired and wireless networks [65]. The raw data is first encoded by the sender using one of the two schemes: MPEG temporal scalable coding and MPEG FGS coding. The encoded video data streams are processed by the sender using the proposed rate control and error control algorithms. After being propagated on an error-prone wireless link, the data streams are processed by the receiver using the proposed error control algorithm. The parameters for error model in (2.4) are $\mu_{gg} = 0.995$, $\mu_{bb} = 0.96$, $e_g = 10^{-6}$. e_b

is chosen to achieve the desired steady state BER from 1% to 10%. Most of the following results are for the case when BER is equal to 5%.

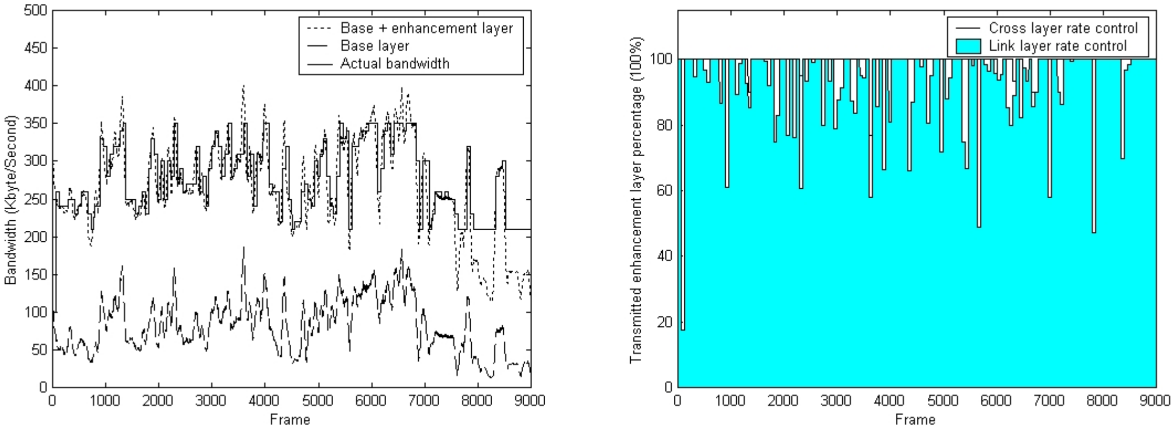
Figure 2.10 and Figure 2.11 show the performance of rate control on MPEG scalable encoded video and MPEG FGS encoded video, respectively. The bandwidths shown include those of the base layer data, the aggregate data (base and enhancement layer), and the actual transmitted data after rate control. The percentage of enhancement layer data that is actually transmitted shows that more enhancement layer data can be accommodated by the proposed cross-layer rate control algorithm than that by link layer rate control.



(a) Base, enhancement layer, and actual transmission bandwidth

(b) Percentage of transmitted enhancement layer data

Fig. 2.10 Rate control for temporal scalable MPEG-4 video

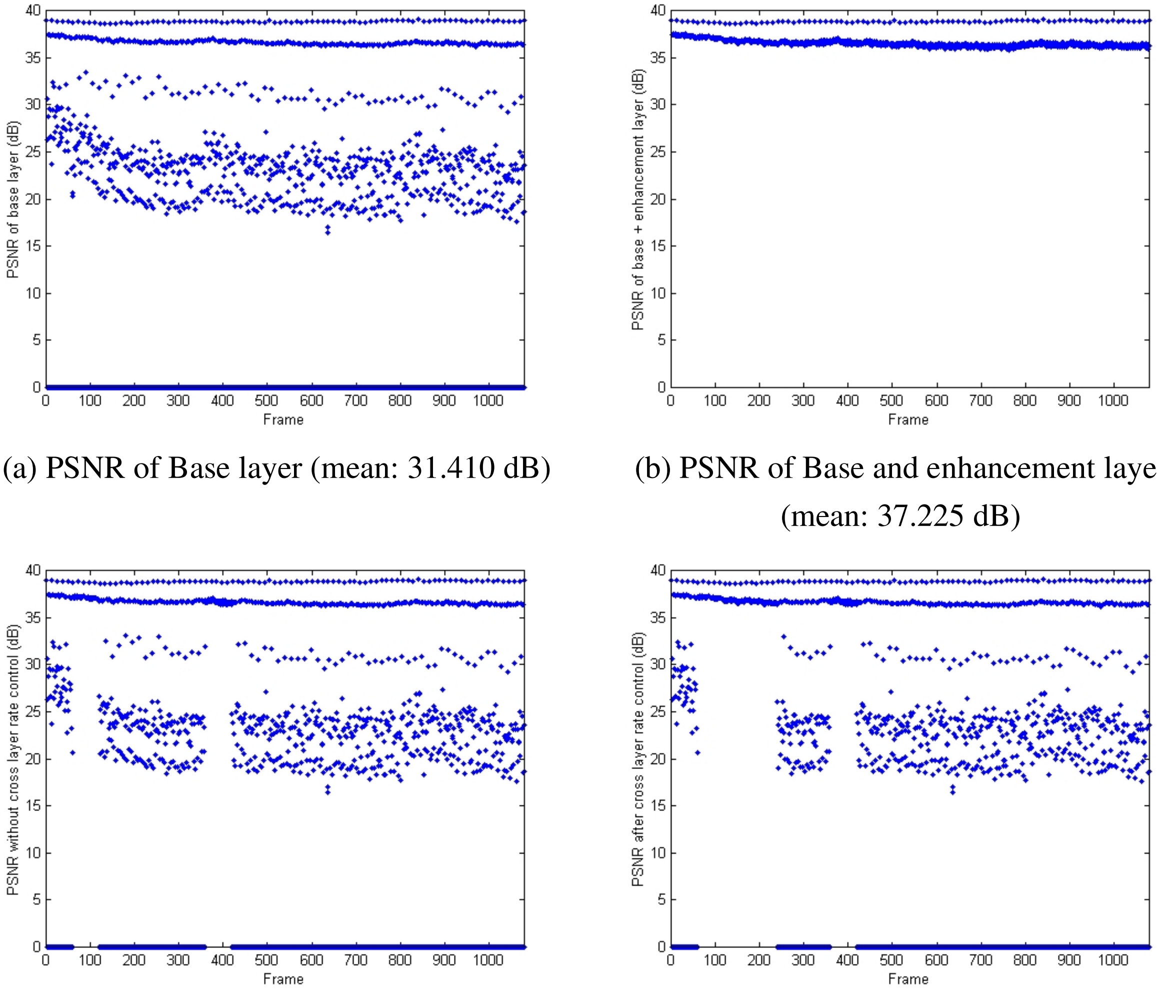


(a) Base, enhancement layer, and actual

(b) Percentage of transmitted enhancement transmission bandwidth layer data

Fig. 2.11 Rate control for MPEG-4 FGS video

The Peak-Signal-to-Noise-Ratio (PSNR) is displayed in Figure 2.12 and Figure 2.13 at the sender's side with rate control and the receiver's side with error control, respectively. The PSNR of transmitted data falls between those of the base layer data and the aggregate data. According to Figure 2.12, it is evidently shown that the cross-layer rate control achieves higher quality. That is, its PSNR is closer to the ideal case when the whole enhancement layer is allowed to be transmitted. Due to packet loss on wireless links, the PSNR of actual received data, as shown in Figure 2.13, is smaller than that of the transmitted data. But error control algorithm helps to get a better quality. The parity data length *R* is also shown in Figure 2.13c, which is adaptive based on the network condition.



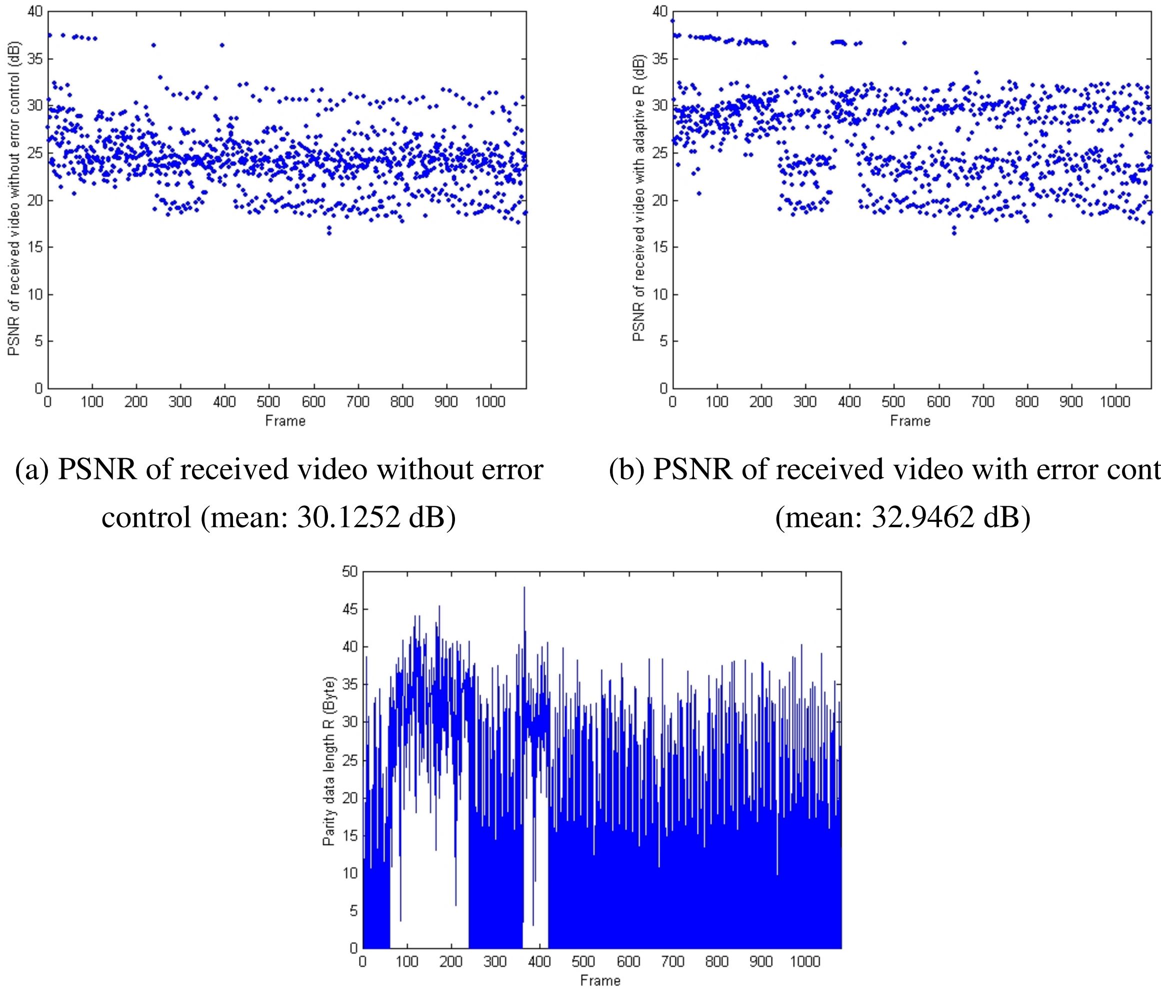
(a) PSNR of Base layer (mean: 31.410 dB)

(b) PSNR of Base and enhancement layer (mean: 37.225 dB)

(c) PSNR of transmitted video with link layer rate control (mean: 35.711 dB)

(d) PSNR of transmitted video with cross-layer rate control (mean: 37.088 dB)

Fig. 2.12 PSNR results for transmitted video (rate control)



(a) PSNR of received video without error control (mean: 30.1252 dB)

(b) PSNR of received video with error control (mean: 32.9462 dB)

(c) Parity data length R (mean: 14.598 bytes)

Fig. 2.13 PSNR results for received video (error control)

The above results are from experiments on movie *War Stars IV* encoded at high quality, Table 2.1, 2.2, and 2.3 further show the experimental results for six different encoding qualities, eight movies encoded by MPEG temporal scalable coding, and three movies encoded by MPEG FGS coding, respectively. They all give the same observations as above.

Table 2.1
Results for different encoding qualities

Quality	Original PSNR (dB)	Link layer rate control		Cross-layer rate control			Cross-layer rate + error control	
		Sent enhancement layer (%)	*Sent PSNR (dB)*	*Sent enhancement layer (%)*	*Sent PSNR (dB)*	*Received PSNR (dB)*	*Received PSNR (dB)*	*Parity data R (Byte)*
High	37.225	80.055	35.711	98.222	37.088	30.1252	32.9462	14.598
Medium	31.795	99.944	31.793	99.944	31.793	28.1804	30.3623	29.665
Low	28.390	99.944	28.389	99.944	28.389	27.0317	27.8962	30.058
256 kb/s	34.524	99.944	34.522	99.944	34.522	30.2827	32.8701	29.240
128 kb/s	33.101	99.944	33.099	99.944	33.099	29.7964	31.8715	29.202
64 kb/s	31.279	99.944	31.278	99.944	31.278	28.5134	30.2560	29.151

Table 2.2
Results for different movies encoded by MPEG scalable coding

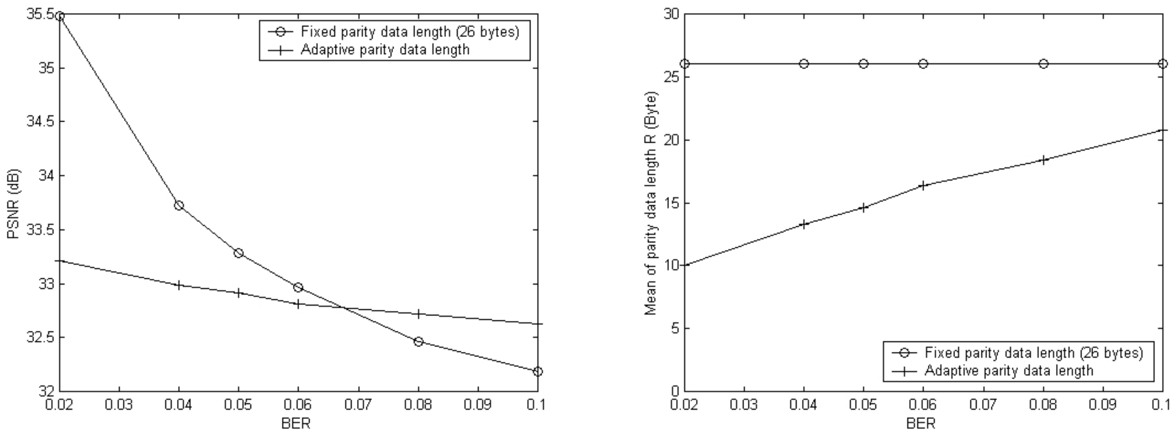
Movie	Original PSNR (dB)	Link layer rate control		Cross-layer rate control			Cross-layer rate + error control	
		Sent enhancement layer (%)	*Sent PSNR (dB)*	*Sent enhancement layer (%)*	*Sent PSNR (dB)*	*Received PSNR (dB)*	*Received PSNR (dB)*	*Parity data R (Byte)*
Star Wars IV	37.225	80.055	35.711	98.222	37.088	30.1252	32.9462	14.598
Citizen kane	37.791	88.333	36.825	97.556	37.564	29.1975	33.7067	29.6640
Aladdin	36.107	28.944	29.761	38.221	30.385	21.8664	25.6563	26.5270
Jurassic Park I	36.661	64.444	33.941	88.444	35.649	28.6769	32.4610	27.2220
Silence of the Lambs	37.975	30.665	33.697	40.887	34.058	29.6585	31.3415	10.0810

Star Wars V	36.595	62.611	34.386	89.000	35.841	26.4372	31.0195	29.2075
The Firm	36.782	76.611	35.065	94.889	36.340	26.6107	31.2513	25.4033
Terminator I	37.199	69.889	34.287	94.889	36.618	26.8716	31.7590	28.8858

Table 2.3
Results for different movies encoded by MPEG FGS coding

Movie	Original PSNR (dB)	Link layer rate control		Cross-layer rate control			Cross-layer rate + error control	
		Sent enhancement layer (%)	*Sent PSNR (dB)*	*Sent enhancement layer (%)*	*Sent PSNR (dB)*	*Received PSNR (dB)*	*Received PSNR (dB)*	*Parity data R (Byte)*
Star Wars	43.832	96.701	42.217	99.803	42.487	25.4646	30.4831	30.0830
The Firm	43.726	95.237	42.117	98.353	42.443	25.8791	30.9449	30.1360
Toy Story	43.956	86.353	41.820	88.517	41.894	25.7340	30.7268	30.1080

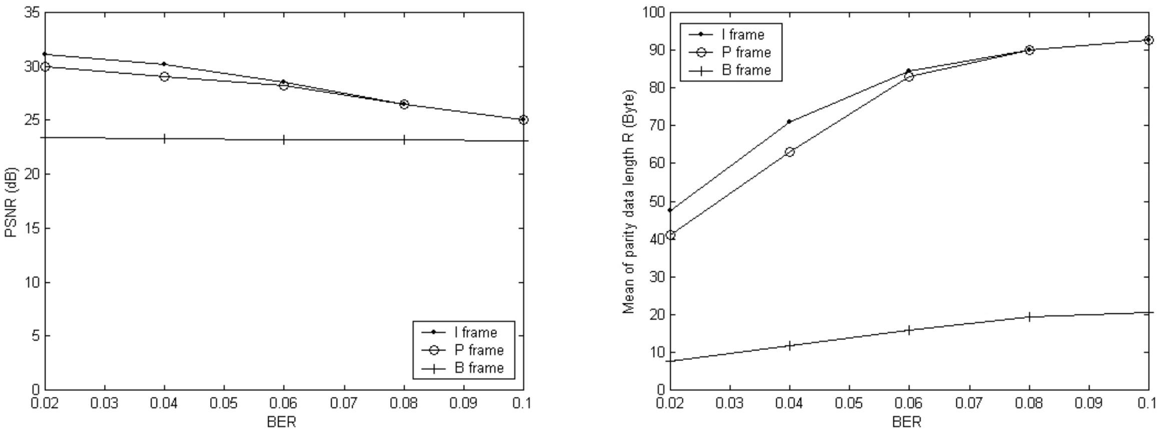
Additional experiments are conducted to test the proposed adaptive error control scheme. In Figure 2.14, it is compared with the scheme using fixed parity data length in the wireless environments with variable BERs. The adaptive scheme makes a good trade-off between error protection and network traffic so that a nearly constant PSNR is obtained for different BERs. The fixed parity data length scheme also works well when the BER is small, at the expense of more communication overhead introduced by extra parity data.



(a) PSNR (b) Mean of parity data length *R*

Fig. 2.14 Comparison of adaptive *R* and fixed *R* error control algorithms

As stated in Section 3.3 (i), the proposed error control scheme can provide multiple-level error protection according to the priority of data streams. The larger value is chosen for parameter *b* in Equation (3) for data with higher priorities. For MPEG-4 video streaming, *I*, *P* and *B* frames can be protected by decreasing error protection capacities, and *b* is chosen as 3, 2 and 1 for *I*, *P* and B frame, respectively. As shown in Figure 2.15, the mean parity data length for *I* frame is the largest, so it has the highest PSNR.



(a) PSNR (b) Mean of parity data length *R*

Fig. 2.15 Error protection of multiple levels

2.5 Summary

This chapter reviews the current research results on real-time data, especially video, transmission over mobile wireless networks and presents new cross-layer error control and rate control algorithms. Inter-layer communication is employed to improve the efficiency of error control and the accuracy of transmission rate assignment. The proposed algorithms are theoretically analyzed. Simulation results demonstrate that, for six videos encoded at different quality levels, the proposed error control algorithm improves the video quality from 0.86 dB up to 2.8 dB, and the proposed rate control algorithm improves the percentage of transmitted enhancement layer data by 15.1% and 0.1% for scalable encoded MPEG video and FGS MPEG video, respectively.

Although many research results have been presented in the past several years to enable smooth video transmission over mobile wireless networks, there are still some open problems in need of future study.

The cross-layer algorithms and protocols for wireless networks are still at their early stage. Some future research topics include cross-layer parameter optimization and integrating cross-layer algorithms to the wireless network specifications.

The multimedia applications often include different kinds of data, such as image, audio and video. It will be challenging to successfully deliver and synchronize the combination of such real-time data over wireless networks.

This chapter considers video transmission over wireless networks. Some video applications, however, involve communications between servers in the wired network and clients in the wireless network. Cheung ([38]) recently introduced an intermediate agent or proxy, located at the junction of backbone wired network and wireless link, to dynamically feedback the network condition in order to help sender make adaptive QoS control. More efforts are needed to investigate the rate control and error control algorithms in such hybrid wired and wireless networks.

Peer to peer communication in mobile ad hoc networks introduces some new and challenging problems, such as how to take advantage of multiple paths between a sender and a receiver, how to control real-time data transmission among heterogeneous devices, and how to deal with the mobility of devices, and so on.