An Improved UDP Protocol for Video Transmission Over Internet-to-Wireless Networks

Haitao Zheng, *Member, IEEE,* and Jill Boyce, *Member, IEEE*

***Abstract*—Packet video will become a significant portion of emerging and future wireless/Internet traffic. However, network congestion and wireless channel error yields tremendous packet loss and degraded video quality. In this paper, we propose a new complete user datagram protocol (CUDP), which utilizes channel error information obtained from the physical and link layers to assist error recovery at the packet level. We propose several maximal distance separable (MDS) code-based packet level error control coding schemes and derive analytical formulas to estimate the equivalent video frame loss for different versions of user datagram protocol (UDP). We validate the proposed packet coding and CUDP protocol using MPEG-coded video under various Internet packet loss and wireless channel profiles. Theoretic and simulation results show that the video quality can be substantially improved by utilizing the frame error information at UDP and application layer.**

***Index Terms*—Forward error correction, packet loss, protocols, video coding, wireless networks.**

I. Introduction

INTERACTIVE and network-based multimedia applications such as video, image, and audio are being used increasingly both in the Internet and over wireless channels. Given the success of digital cellular networks, it is inevitable that future wireless services will support Internet Protocol (IP)-based multimedia applications [1], [2]. Typical applications include mobile internet access, mobile videoconferencing, streaming video/audio, distance learning, e-commerce, entertainment, etc. Particularly, in an Internet-to-mobile traffic flow scenario, as shown in Fig. 1, the multimedia packets are first sent through Internet and then over wireless packet networks. Most Internet-based real-time multimedia services employ user datagram protocol (UDP) as their transport protocol [3]. Compared to transmission control protocol (TCP) [4], UDP does not yield retransmission delay, which makes it attractive to delay sensitive applications. A UDP packet consists of a header and payload. UDP employs a cyclic redundancy check (CRC) to verify the integrity of packets; therefore, it can detect any error in the packet header or payload. If an error is detected, the packet is declared lost and discarded. UDP packet transmission in Internet is "best effort," in which case network congestion yields packet loss. At the receiving host, packets are either perfect or completely lost.

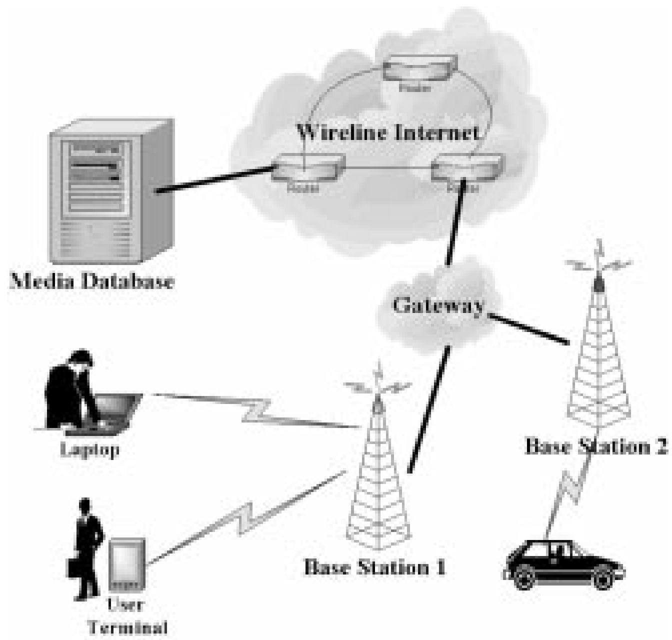


Fig. 1. Internet-to-wireless multimedia communications.

In contrast, wireless packet networks are characterized as low-bandwidth and unreliable, in which a considerable amount of packet losses are induced by both channel failure and network congestion. Depending on the environment, moving speed, and network loading, packet loss can be random or bursty. Since UDP does not perform any error recovery, streaming multimedia over wireless networks can yield unpredictable degradation and poor video/audio quality. One inefficiency of UDP is that it fails to incorporate the properties of the wireless network, where a channel error only partially corrupts a packet. UDP discards a packet containing only a small part of corrupted data. As such, it also throws out error-free data within the packet. Indeed, the current and emerging multimedia coding technologies are focusing on providing error resilience so that the media decoder can tolerate a certain amount of channel errors. To support this feature, wireless systems should revise the UDP protocol to reduce or avoid unnecessary packet discarding.

Reliable UDP (RUDP) was proposed to provide reliable in-order delivery up to a maximum number of retransmissions for virtual connections [5]. RUDP can calculate the CRC based on packet header or header plus payload. This flexibility makes it suitable for transport telecommunication signaling. UDP Lite

Manuscript received September 1, 2000; revised March 1, 2001. The associate editor coordinating the review of this paper and approving it for publication was Dr. John A. Sorensen.

H. Zheng is with Bell Laboratories, Lucent Technologies, Holmdel, NJ 07733 USA (e-mail: haitaoz@lucent.com).

J. Boyce was with with Bell Laboratories, Lucent Technologies, Holmdel, NJ 07733 USA. She is now with Thomson Multimedia Corporate Research, Princeton, NJ 08544 USA.

Publisher Item Identifier S 1520-9210(01)07788-4.

protocol [6] was proposed to prevent unnecessary packet loss at the receiver if channel errors are located only in the packet payload. The CRC is constructed based on packet header, so that only corrupted packet headers result in packet loss. UDP Lite delivers packet payload, whether perfect or erroneous to the upper layers.

For a packet switched network that exhibits a fairly high packet loss rate (PLR), employing forward error correction (FEC) code to the application packets provides an effective way to mitigate channel unreliability and improve media quality [7]–[9]. These techniques are currently being considered by the IETF for supporting real-time multimedia communications in Internet and over wireless networks. Certain number of lost packets can be recovered by applying maximal distance separable (MDS) codes, i.e., Reed–Solomon (RS) codes, across the packets [7]–[12]. For example, the encoder chooses k information packets and generates $n-k$ parity packets to construct an (n, k) RS codeword. For Internet-based transmission, the packets are numbered and are assumed to arrive perfectly or never arrive at all. The receiver can recognize the missing packets and replace them as erasure packets. Since an (n, k) RS code can correct $(n-k)$ erasures, this packet coding scheme can recover up to $(n-k)$ packet losses [7]. The MDS codes are systematic so that if all the k information packets are received, the receiver can bypass the parity packets, or upon receiving any k packets, it can start the decoding process to recover the lost information packets. As such, the delay is reduced to minimum.

Within wireless networks, the UDP packets that are corrupted by channel errors would be discarded and thus deemed as erasure packets. The above (n, k) packet coding can be applied to recover the packet losses. Since any error within a packet would erase the whole packet, even a small physical layer error can yield a high PLR. Therefore choosing an appropriate coding rate n/k depends on the physical layer performance and the length of the packet. Large packets require large number of parity packets to effectively mitigate the information. This further increases the overhead and end-to-end delay, as well as complexity. On the other hand, when the system employs UDP Lite, the packets that are corrupted but have valid headers can still be forwarded to the FEC decoder. In this case, the FEC decoder performs both error correction and erasure recovery. It should also be pointed out that the MDS codes are twice as powerful in erasure recovery compared to error correction, i.e., they can recover up to $n-k$ erasures or up to $(n-k)/2$ errors. As such, UDP Lite does not utilize the FEC coding to its full effectiveness. These analysis points out the need for an improved UDP protocol that supports FEC coding at the packet level to effectively reduce information loss. The existing UDP and UDP Lite protocols fail to incorporate all the channel information from physical layer. In this paper we propose an improved UDP protocol that captures the frame error information to assist packet level error recovery. One immediate application of this design is streaming multimedia services over wireless packet networks.

This paper is organized as follows. Section II briefly describes the protocol stack in wireless links and how UDP performs over wireless links. We then revise the UDP protocol to capture the channel error information from the physical layer in Section III. The improved version is named a *complete user datagram protocol (CUDP)*. In Section IV, we address the problem of FEC design on application packets that uses the physical layer information to achieve effective packet recovery. In Section V, we characterize the performance of UDP, UDP Lite, and CUDP in terms of the probability of decoder failure, under different network and channel conditions. The analytical results determine the potential benefit of CUDP by making physical frame error information available to the FEC decoder. In Section VI, the comparison among different UDP protocols is established by simulating MPEG video over wireless networks using both theoretical channel model and real channel traces. Finally, we conclude the paper in Section VII.

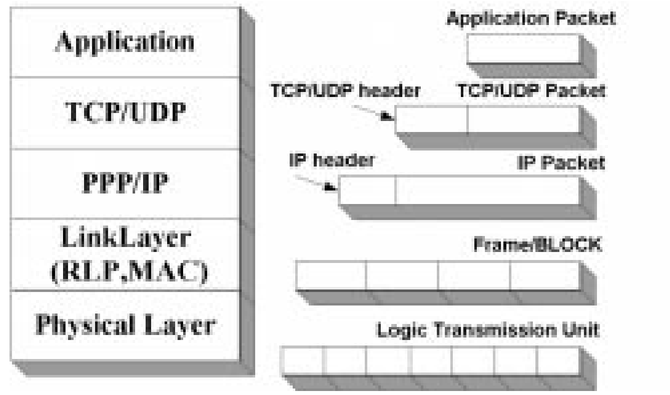


Fig. 2. General protocol stack and packet structure.

II. UDP OVER WIRELESS LINKS-EXISTING DESIGNS

Fig. 2 illustrates a general wireless protocol stack and data unit associated with each layer. After attaching UDP(TCP)/IP/PPP related headers, the application packets are deemed as a continuous bit stream at the link layer. To accomplish physical transmissions that are burst by burst, the link layer partitions the packets into multiple units. The unit size depends on the configuration of radio link protocol (RLP), medium access control (MAC), and physical (PHY) layer, as well as the current channel status, but is usually small compared to the packet length. In third-generation (3G) wireless systems [1], [2], for applications that requires low and medium data rates, each physical layer frame corresponds to a transmission unit. To support high data rate services, MAC protocol specifies that RLP can subdivide each physical layer frame into smaller logical frames named logical transmission units (LTUs), each associated with a 16 bits of CRC [13]. Typical LTU size can vary from 300 to 600 bits (40–80 bytes), while IP packets are typically 600–1500 bytes long. In the remainder, we simply use frame to represent both frame and LTU.

At the MAC/PHY layer, channel coding is applied to each frame to protect the information data. While at the receiver, residue error after channel decoding can be detected using CRC. This frame error information is available at the RLP layer. It should be noted that in a time-varying channel, the transmitter could adjust the format of channel coding and modulation in each frame, i.e., apply link adaptation to maintain quality of service (QoS) requirements. It is possible to combine link adaptation with FEC coding at the packet level to achieve maximum flexibility. However, this approach suffers from a significant level of signaling, delay and complexity. In the remainder, we assume that packet FEC coding at application layer is performed for a given bandwidth and channel FER requirement. And link adaptation is employed to maintain such requirement. The relation between these two designs is absorbed in the definition of data rate and channel FER.

While satisfying the delay requirement, the RLP layer at the receiving host can specify a limited number of retransmissions to compensate for frame losses. However, such an error handling procedure can not guarantee error free delivery so some frames would still be corrupted. We assume that the benefit of retransmissions is embedded in the FER and thus is irrelevant to our protocol and FEC coding design. The RLP forward the received frames to the point-to-point protocol (PPP) [16] for packet reconstruction. In current wireless systems, the erroneous frames are not forwarded to PPP or its equivalent layer and there is no indication of missing frames. This yields packet loss. When TCP is employed, packet loss can be recovered through congestion control. The performance of TCP/RLP was studied in [14] and [15].

As explained before, UDP does not perform any error recovery. Upon receiving a packet, UDP performs CRC to validate the packet, including both packet header and payload. In this case, any frame loss would result in the whole packet being discarded. Mathematically, the PER can be approximated as

$$\mathrm{PER} = 1 - (1-p)^m \approx mp \tag{1}$$

where m represents the number of frames per packet and p represents the residue frame error rate (FER) after channel coding and retransmission. PER grows linearly with the packet length and FER. A typical 1% FER and ten frames per packet would yield a PER of 10%. As we pointed out in Section I, UDP discards a partially corrupted packet so that the error-free data is wasted.

UDP Lite is superior to UDP by forwarding the corrupted data packets to the FEC decoder [6]. However, if the erroneous frames are discarded by link layer, the frames within the packet are misplaced. This can generate additional but unnecessary data loss. On the other hand, even when the link layer is configured to forward all the frames, error-free and corrupted, to the upper layers, the locations of the corrupted data units are unknown. UDP Lite does not consider the usage of CRC in each wireless frame, which provides frame error indication. When the frame size is sufficiently small compared to the packet length, such an indication still provides a reasonable estimation of the error locations.

III. An Improved UDP Protocol Design

The previous sections show that UDP and UDP Lite failed to provide the most efficient packet transmission over wireless networks due to the ignorance of frame error information from the link layer and physical layer. We propose to exploit this information to improve the transmission efficiency. The improved design includes two stages. First, it is known that the current protocol stack design does not support information communications from RLP layer to PPP/IP/UDP and application layers. Therefore, we propose to redesign the interface between RLP and PPP, PPP and IP, IP and UDP, so that certain information can be exchanged in both directions [17]. In addition, the redesigned RLP should forward the corrupted frames to the PPP or equivalent layer. Second, the improved UDP should apply CRC to the packet header only and forward the packet payload to the application. It should also organize the frame error information to a format that is understandable by the application. The format of error information depends on the system implementation as well as the application. We illustrate two example formats for applications invoking FEC coding.

- *Type 1:* LTU Error Indicator (For FEC decoders that require erasure indicator)

  The frame error information is represented in terms of a set of error indicators that are associated with each packet. The error indicators contain the starting and ending location of the erroneous frame. If the packet header is valid, UDP forward the indicator and the packet payload to the FEC decoder.
- *Type 2:* Reformatted Packet (For FEC decoders that can recognize erasures)

  The frame error information is incorporated within the packet payload. In this case, if a physical frame is corrupted, the payload is represented as a set of erasures, which can be recognized by the FEC decoder. The erasure format depends on the system implementation. Under a valid packet header, UDP passes the reformatted packet payload to the upper layers.

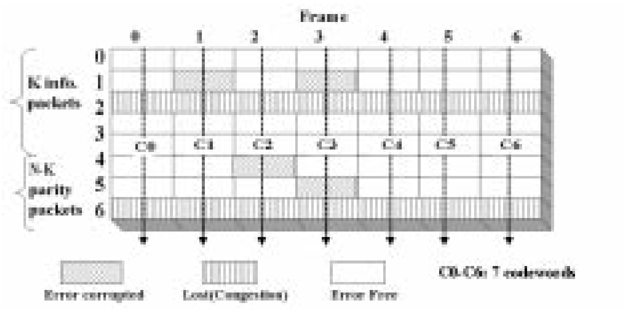
We refer to the proposed UDP design as *complete UDP (CUDP),* since it captures all the available information, i.e., the error-free frames and the location of erroneous frames. When combined with FEC coding, it turns erroneous frames into erasure frames so that the other error-free frames within the same packet can be utilized to recover the information loss. When there is no FEC coding, forwarding the error location to application still benefits the overall performance. For video and audio in particular, the corrupted frames can be forced into an all “1” sequence, so that media decoder can recognize this invalid sequence, and invoke error concealment to reduce or sometimes eliminate the error effect.

IV. Packet Coding Design

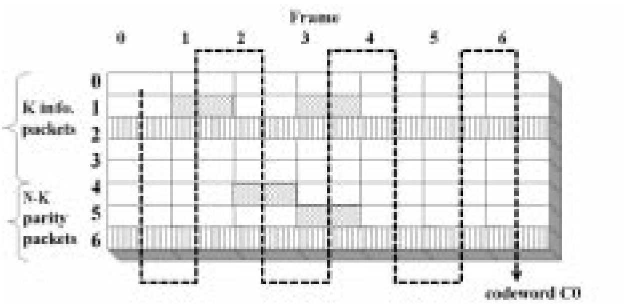
We propose two FEC coding schemes at the packet level to take the advantage of the wireless frame error information proposed in the previous section.

A. Vertical Packet Coding (VPC)

The FEC encoder picks k packets and applies FEC coding across the packets. For real-time applications, the packets that are coded together should have the same or similar delay constraint. Applications like streaming video/audio and video-conferencing can group the packets within the same video frame together since they have the same delay requirement. Multiple



(a)



(b)

Fig. 3. (a) Vertical packet coding (VPC). In this example, four information packets are encoded together to generate three parity packets. (b) Long vertical packet coding (LVPC). In this example, four information packets are encoded together to generate three parity packets.

video frames can be grouped together to increase the MDS code (n, k) values, which improves burst error resiliency, but this also increases the delay which depends on the amount of data sent in k packets.

In order to generate the parity packets, the information packets should have the same length and if not, they are bit stuffed to match the longest one. In other words, the length of the parity packets is equal to that of the longest information packet. Since the stuffing bits in the information packets are for computational purposes only, and not transmitted over the air, this coding scheme does not cost additional overhead. Alternatively, the source coding and packetization scheme can be designed to generate packets of equal or similar size [17].

Fig. 3(a) illustrates the transmitter structure. The channel encoder at the application layer takes one data unit from each of k packets and generates $(n-k)$ parity units to construct $(n-k)$ additional packets. We name this coding scheme *vertical packet coding (VPC).* VPC provides transparent Internet-to-Wireless communications. As such, the UDP protocol within the Internet remains unchanged.

The other advantage of CUDP is that even if the VPC decoder fails, some of the erroneous packets can still be recovered. Using the scenario in Fig. 3(a), packets 1, 2, 4, 5, and 6 are declared lost if only the packet CRC check is used to validate the data. Using (7, 4) MDS code, the decoder can only recover three erasures. Therefore, without the frame error information, the conventional decoder will fail. If the frame error information is available, the erasures at columns 0, 1, 2, 4, 5, and 6 are recovered. Only the column corresponding to frame 3 contains erasure. For compressed multimedia data, larger amount of correct information leads to better error concealment and recovery, therefore higher media quality. Using the same scenario to evaluate the performance of UDP Lite with (7, 4) VPC code, the decoder can recover one error and one erasure, or three erasures; therefore it can only recover columns 0, 4, 5, and 6.

B. Long Vertical Packet Coding (LVPC)

For a fixed redundancy ratio $(n-k)/n$, MDS codes achieve better error/erasure correction efficiency as n increases, at the cost of increased computation complexity. Assuming the information packets are of length X, the FEC encoder can increase n by coding L multiple columns of data units together and generate X/L MDS (nL, kL) codewords, as opposed to the VPC method which generates X MDS (n, k) codewords. The delay is not increased over the VPC method because the delay is still based on k packets. We refer to this coding scheme as *long vertical packet coding (LVPC),* as shown in Fig. 3(b). Assuming $L = m = 7$, where m represents the number of frame per packet, the dimension of the MDS code becomes (49, 28), and the erasure recovery capability increases to 21. Assuming the same error pattern in Fig. 3(a), the decoder can recover all the erasures. This coding efficiency is obtained at the cost of increased decoding complexity. It would also require the transmitter to have access to the wireless frame size. In addition, if the decoder fails, all the erasures can not be recovered, while for VPC, some of the erasures can be recovered.

V. Analytical Performance

To quantify the performance of the UDP protocols and FEC coding schemes, we use the probability of decoding failure as the metric. It represents the data loss rate from the application's point of view. The decoder fails when the errors/erasures outnumber the FEC error/erasure correction capability. For a group of packets coded together, the decoding failure can be defined as the *group of packet error rate (GPER).* If the group belongs to one video frame, GPER corresponds to the video frame loss rate.

A. Performance Analysis for Wireless Packet Flow

We begin our study by assuming information loss only happens in the wireless network, and looking at the GPER of UDP, UDP Lite, and CUDP as a function of the FER, hereby denoted as p, the MDS code parameters n, k and the packet length which is represented by the number of frames per packet m. We use Per to represent the packet loss rate and $Per = 1-(1-p)^m$.

UDP+VPC: For conventional UDP and (n, k) MDS codes, the rate of decoder failure $P_{\mathrm{UDP}}(p, n, k)$ corresponds to the probability of more than k packets within n packets are corrupted, i.e.,

$$\begin{aligned} P_{\mathrm{UDP}}(p,\, n,\, k) &= \sum_{i=n-k+1}^{n} \binom{n}{i} Per^i (1-Per)^{(n-i)} \\ &= \sum_{i=n-k+1}^{n} \binom{n}{i} (1-(1-p)^m)^i (1-p)^{m(n-i)}. \qquad (2) \end{aligned}$$

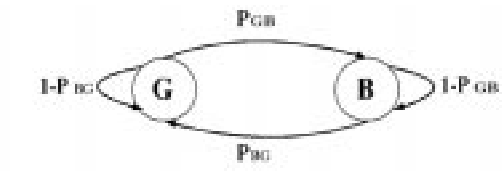


Fig. 4. Two-state channel model for bursty error.

CUDP+VPC: In this case, the decoder fails if there are more than k corrupted or lost frames within any single column, hereby represented as

$$P_{\rm CUDP}(p,\,n,\,k) = 1-\left(1-\sum_{i=n-k+1}^{n}\binom{n}{i}p^i(1-p)^{n-i}\right)^m. \quad (3)$$

It should be noted that for both UDP and CUDP, the performance depends on the error statistics at the physical frame level. Since all the data within a corrupted frame are declared as erasures, the error pattern within the frame has no impact on the decoder performance. This conclusion is based on the assumption that the error burst length is small enough compared to the frame length, so that if an error burst occurs, it is unlikely to affect more than one frame.

UDP Lite+VPC: In this case, the decoder fails if there are two erroneous data units (hereby assume bytes for MDS code based on $GF = 2^8$) in the same column. Therefore, the performance depends on the error pattern within each frame. To simulate a wireless channel with various burst error occurrences, we use an analytically tractable two state Gilbert–Elliot model. Accordingly, the model has two states, good (G) and bad (B). The bits are received correctly in good state while being corrupted in bad state. As shown in Fig. 4, the transition probabilities between the two states $P_{\rm GB}$ and $P_{\rm BG}$ fully represent the error model.

We assume that the physical layer frame contains S bytes information and the wireless channel yields an average error burst of B bits. Again, we assume that the burst length is small enough compared to the frame length, so that the error events are independent from frame to frame. In this case, the decoder fails when there are more than $(n-k)/2$ corrupted bytes within the same column. And probability of such event is expressed by

$$P_{lite}(p,\,n,\,k,\,S) = 1-\left\{1-\sum_{i=(n-k)/2+1}^{n}\binom{n}{i}u^i(1-u)^{n-i}\right\}^{mS} \quad (4)$$

where u represents the byte error rate, which can be derived as a function of $P_{\rm GB}$ and $P_{\rm BG}$. As can be seen, the performance depends on both the packet length mS and the frame length S.

As for our sample design, we use a $(n, k) = (8,\,6)$ MDS code based on $2^8\,GF$ and assume $m = 5$. A performance comparison in terms of GPER corresponding to 80 bytes frame size is presented in Fig. 5. These results, using VPC, show that CUDP effectively reduces GPER compared to UDP and UDP Lite. The

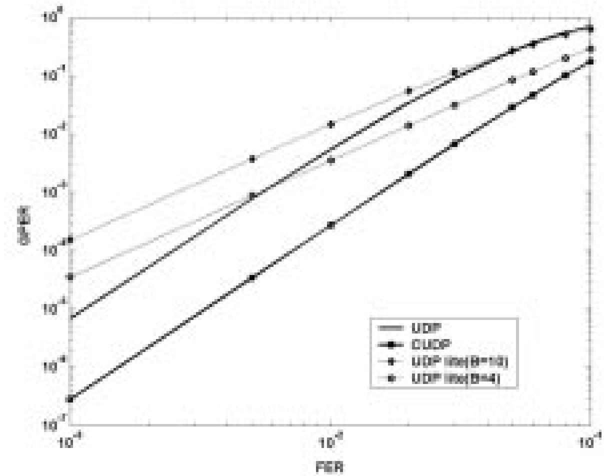


Fig. 5. Group of packet error rate (GPER) for wireless channels using VPC. System configuration: MDS (8, 6) code with error or erasure decoding, five frames per packet, 80 bytes frame size. Wireless channel is generated using Gilbert–Elliot model with average burst error length $B = 4$ and 10 bytes.

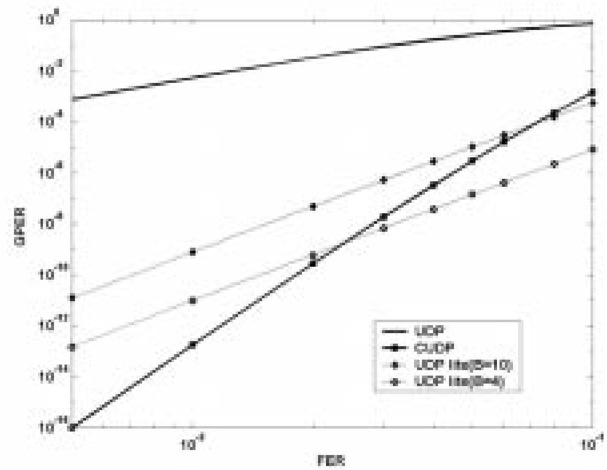


Fig. 6. GPER for wireless channels using LVPC packet coding scheme. Gilbert–Elliot wireless model with average burst error length $B = 4$ and 10 bytes. MDS (8, 6) code with five frames per packet.

performance of UDP Lite is also illustrated in the figure, for different values of the Gilbert–Elliot model parameter $P_{\rm GB}$, such that the average burst error lengths are 4 and 10 bytes. For a given FER, the burst length heavily impacts the performance of UDP Lite, and the difference remains constant regardless of the FER. On the other hand, based on the assumption that the error burst remains in the same frame, for a given FER, the burst error length, which represents the error pattern, does not affect the results of UDP and CUDP.

The protocol performance with LVPC can be derived similarly.

The GPER versus FER response of a (8, 6) MDS coding design is shown in Fig. 6. CUDP achieves significant GPER improvement compared to that of UDP and UDP Lite. With respect to UDP Lite, reduced burst length will translate into a decoder gain. Once the FER grows to 3% and higher, UDP Lite even outperforms CUDP. In this case, although many frames

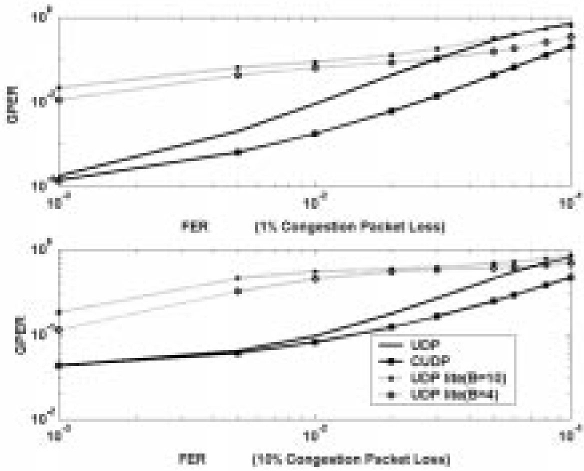


Fig. 7. GPER for hybrid Internet-to-wireless network, using VPC as packet coding scheme. Gilbert–Elliot wireless model with average burst error length $B = 4$ and 10 bytes. MDS (8, 6) code with five frames per packet. Random Internet packet loss rates $q =$1% and 10%.

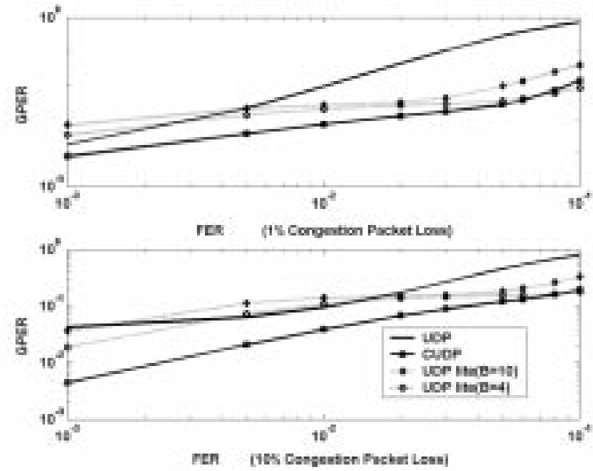


Fig. 8. GPER for hybrid Internet-to-wireless network, using LVPC as packet coding scheme. Gilbert–Elliot wireless model with average burst error length $B = 4$ and 10 bytes. MDS (8, 6) code with five frames per packet. Random Internet packet loss rates $q =$1% and 10%.

are corrupted, UDP Lite exploits the error-free bits/bytes within the packets to recover the erroneous bits/bytes. For a MDS code of large dimension, this appears to be more effective than marking the whole frame as erasures. It is also observed that using LVPC, UDP Lite is superior to UDP, as opposed to that shown in Fig. 5 where it is true only for large FER. Since UDP declares the whole packet as erasure, the number of erasures scales linearly with the codeword length. Therefore, when combined with UDP, LVPC fails to increase the erasure recovery capability.

B. Performance Analysis of Internet-to-Wireless Packet Flow

Next, we extend the analysis to a hybrid Internet and wireless network. It is difficult to construct a mathematical model that captures all the characteristics related to wired Internet packet loss, so we simply assume that the Internet packet losses are random with a uniformly distribution of rate q. As in the previous section, the Gilbert–Elliot burst loss model is used for the wireless link. Accordingly, the decoding error probability for UDP can be expressed as

$$\begin{aligned}\hat{P}_{\mathrm{UDP}}(p, q, n, k) &= \sum_{j=0}^{n-k} \binom{n}{j} q^j (1-q)^{n-j} \cdot P_{\mathrm{UDP}}(p, n-j, k) \\ &\quad + \sum_{j=n-k+1}^{n} \binom{n}{j} q^j (1-q)^{n-j} \end{aligned} \tag{5}$$

and the performance of CUDP and UDP Lite can be derived similarly.

In Fig. 7, we validate the performance of UDP, CUDP, and UDP Lite for $n = 8$, $k = 6$, and uniformly distributed random packet loss rate of 1% and 10%. We observe that the importance of frame error location diminishes as the Internet packet loss rate grows. When the wireless network exhibits higher stability compared to the Internet, for example, at a packet loss rate of 10% and FER of 0.1%, the performance of UDP and CUDP are quite close. When FER grows to 1% and higher, CUDP outperforms UDP in a noticeable manner. It is also observed that large error burst has negative impact on the performance of UDP Lite. The performance of LVPC with congestion related Internet packet loss could be computed similarly. Fig. 8 plots the comparison of CUDP, UDP, and UDP Lite using LVPC for $q =$1% and 10%. We apply (8, 6) MDS codes with $m = 5$ frames per packet. Still, CUDP outperforms UDP and UDP Lite in most cases, although for 6% and higher FERs, UDP Lite with LVPC has the best performance, since error correction is more effective for high FER environments.

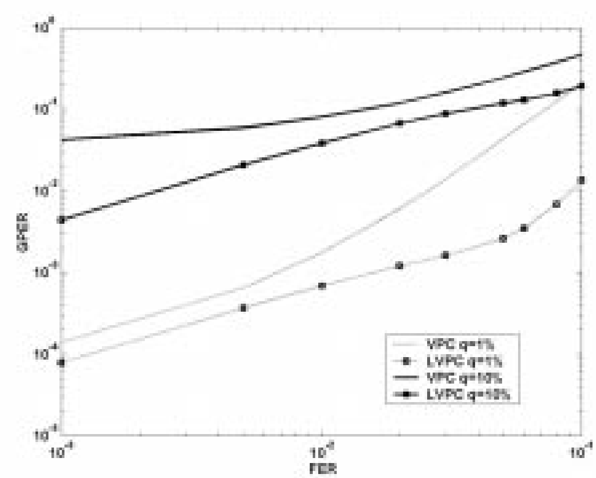


Fig. 9. GPER for hybrid Internet-to-wireless network, using CUDP combined with both VPC and LVPC as packet coding scheme. Gilbert–Elliot wireless model with average burst error length $B = 4$ and 10 bytes. MDS (8, 6) code with five frames per packet. Random Internet packet loss rates $q =$1% and 10%.

We are also interested in comparing the performance of VPC and LVPC employing CUDP. Fig. 9 illustrates the GPER performance under the same configuration as above. LVPC achieves huge performance improvements especially for medium to high FERs and low congestion packet losses.

For 1% congestion loss and 1% FER, LVPC reduces GPER from 0.0001 to 0.000 01; while for 10% FER, 0.06 to 0.0002, compared to VPC. However, LVPC requires the knowledge of frame or LTU size at the encoder, which leads to additional complexity and signaling delay. In addition, as the congestion loss rate increases to 10%, the difference between VPC and LVPC diminishes. Therefore, VPC has more practical importance compared to LVPC.

VI. Application to MPEG-Based Packet Video Over Wireless Networks

The previous section presented analytical performance of the CUDP, UDP, and UDP Lite protocol in terms of GPER. In this section, we evaluate the protocol performance for steaming video applications, by measuring the peak signal-to-noise ratio (PSNR). The MPEG video coding standard was used, and each group of packets contains a single MPEG video frame, so the GPER corresponds to a video frame error rate. Because MPEG uses inter-frame coding, an error in a single video frame can propagate into many decoded video frames, causing long-lasting visual impairment. So the PSNR, which considers the quality of all decoded video frames, provides a more meaningful measurement of video quality than the GPER.

An MPEG video sequence was coded, at a bit rate of 288 kb/s, QSIF (176×120 pixels), and 24 video frames per second. In addition, the HiPP method [18] was used to provide unequal error protection (UEP) for the video, with an overhead rate of 25%, yielding a total transmission rate of 384 kb/s. In the HiPP method, a standards-compliant MPEG video stream is split into high priority (HP) and low priority (LP) partitions, using a technique similar to MPEG-2 data partitioning (DP). The HP data contains the most important information, and video can be decoded, with reduced quality, using only the HP data. Packets are formed which contain the interleaved HP and LP data.

The HP data only was protected with a MDS code, using the VPC method. This consideration aims to balance the tradeoff between overhead and error robustness. UDP, UDP Lite, and CUDP were used to stream the video data, using the VPC method. At the receiver end, MDS decoding was used to correct transmission errors in the HP data, and then the HP and LP partitions were merged into a single MPEG compliant bitstream, which was sent to a standard MPEG decoder. For more information about the HiPP method, see [18]. The following assumptions were made:

- All the packets belong to the same video frame are encoded together. For each frame, the packets have the same length. Therefore, the (n, k) dimension varies from frame to frame, especially between I-frame and P-frame. The packet header contains the (n, k) information so that the receiver can recognize the encoding format. In fact, the (n, k) values are chosen from [18] which generates a 25% overhead.
- Only VPC is simulated. If the VPC decoder failed to recover the packet group, the corrupted packets are deemed as lost and removed without being forwarded to video decoder. This helps to clearly identify the benefit of frame error indication.

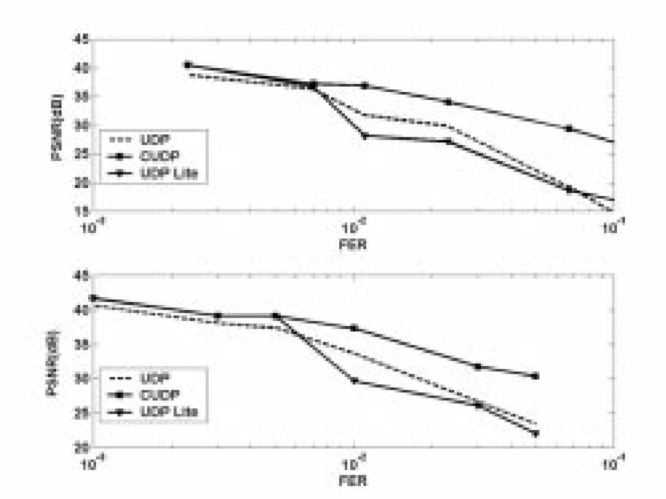


Fig. 10. Video PSNR for Internet + wireless networks with coding VPC. Random Internet packet loss rate of 1% and 10%. Gilbert–Elliot wireless model with average burst error length $B = 4$ and 10 bytes. (a) Congestion packet loss rate $= 1\%$.

- The video sequence contains 1003 video frames. We choose the average PSNR of all the frames to be the performance metric.
- The maximum length of the application packets is limited to 800 bytes.
- Simple video error concealment was used, using motion vector estimation [20]. If the motion vector for a macroblock was unavailable because of transmission errors, an attempt was made to estimate the motion vector based on neighboring macroblocks from the above row. If the above row of macroblocks was available, the estimated motion vector was the median of the three neighboring macroblocks from the row above: 1) above and one macroblock to the left; 2) directly above; and 3) above and one to the right. If the above row of macroblocks was not available, the estimated motion vector used that of the same macroblock position from the previous frame, if it was available. If this was not available, an estimated motion vector of zero was used.

A. *Video PSNR Performance in Theoretical Channels*

We first simulate the video performance where theoretical models were employed to generate the network and channel impairments. The Internet packet loss is modeled as a random event with uniform distribution. The average packet loss rate varies from 1% to 10%. The wireless error traces are generated using the two-state Gillbert–Elliot model while varying the frame error rates and burst lengths. The error traces represent the link level performance that takes into effect of physical layer channel coding and RLP layer retransmission. Each wireless frame contains 90-byte information data and 16-bit CRC check.

The video PSNR corresponding to 1% and 10% Internet packet loss and average burst of 4 and 10 bytes are shown in Fig. 10 as a function of FER. The simulation shows that CUDP utilizes the frame error information to recover corrupted frames and consistently yields an overall good performance. As FER increases, it shows a graceful drop in video quality. We observe 5–10 dB PSNR improvement for 1% congestion loss. At the same time, the PSNR improvement when using CUDP diminishes when the FER is reduced to 1% and lower, even though Fig. 7 shows that the theoretical GPER is significantly lower. For this range of operating points, the information loss becomes relatively smaller so that error concealment techniques can effectively reduce and even eliminate the effect of channel error. Without error concealment, we would expect CUDP to further improve the PSNR at these FERs compared to UDP and UDP Lite.

B. *Video PSNR Performance Using Experimental Channel Traces*

In this section, results are presented for experimental IP packet loss traces. Because of the time-varying nature of Internet packet loss characteristics, it is difficult to make an experimental apples-to-apples comparison of the performance of UDP, CUDP, and UDP-Lite. Rather than independently transmitting each method's separate packet stream over IP for comparison, traces were made of sample packet loss patterns, and then the same loss traces were applied in both cases. The IP packet loss traces were generated by repeatedly transmitting a sample MPEG video clip at a 384 kb/s rate and 800 bytes packet size from a Lucent Technologies facility in Swindon, U.K., to a Lucent facility in Holmdel, NJ. A subset of the traces was selected for use in the experiments to provide a range of packet loss rates. The choice of what packet loss rates to use in the experiments was limited to selecting from among those rates actually observed in the experimental traces. The wireless error traces are obtained under several system configurations including a baseline system employing one transmit and one receive antenna, and BLAST system with two transmit and two receive antennas [19]. In the simulations, each wireless frame contains 180 bytes (1440 bits) information payload. We further partition each frame into two subframes of 90 bytes (720 bits), with separate CRCs.

The impact of the wired Internet congestion packet loss on the average PSNR performance is shown in Fig. 11. We employ a (2, 2) BLAST system that performs at a 4.8% subframe error rate (SFER) and an average burst of 4 bytes. CUDP achieves 2–6 dB of PSNR improvement over that of UDP and 5–10 dB over that of UDP Lite. As congestion packet loss increases, the improvement shrinks, as expected. In Fig. 12, we evaluate the PSNR performance by fixing the congestion packet loss rate and adjusting the value of SFER. When SFER becomes less than 0.5%, the performances of these three protocols are quite similar. For congestion packet loss rate of 0.98% the difference is between 1 and 3 dB, while for 9.8%, the difference reduces to about 1 dB. For small SFER and congestion loss, although CUDP can reduce the number of decoder failure, the picture loss due to wireless error is still small and can be adequately dealt with using the HiPP UEP and error concealment techniques. Therefore, the improvement of CUDP is less perceivable. When congestion loss becomes the dominant impairment, we see much less advantage of CUDP. On the other hand, as SFER grows to 1% and higher, CUDP shows dramatic improvements in PSNR.

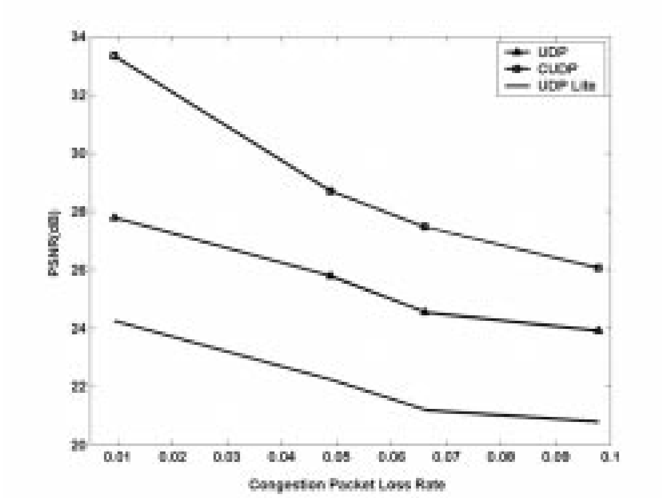


Fig. 11. Video PSNR for Internet + wireless networks with coding VPC, using experimental Internet packet loss traces, and BLAST architecture wireless system.

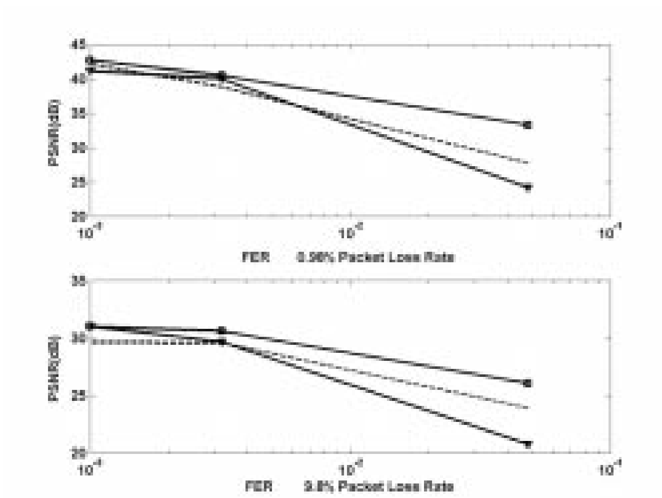


Fig. 12. Video PSNR for Internet + wireless networks with coding VPC, using experimental Internet packet loss traces, with packet loss rates of 0.98% and 9.8%, and CDMA wireless systems.

VII. Conclusion

This paper explores the idea of using channel frame error information to assist error recovery at the applications layer. One immediate application is to accommodate Internet-to-wireless video traffic. We propose a new protocol stack design, which allows bi-directional information exchange so that the physical, link layers can communicate with the application layer. We also propose to improve UDP protocol so that the physical frame error indication is forwarded to the application for better error control. This indication is very valuable when FEC coding is applied to the application packets. We then quantify the theoretical performance of the proposed CUDP protocol and the existing UDP and UDP Lite protocols, in terms of the probability of FEC decoding failure. It was shown that CUDP could more effectively recover from Internet packet losses and corrupted wireless frames than the conventional UDP and UDP Lite protocols, at reasonable packet loss rates and wireless FERs. This theoretical conclusion is further validated by simulating a MPEG video in an Internet-to-wireless packet flow using both analytical channel model and experimental error traces. We summarize the finding as follows:

- CUDP provides great flexibility for applications to utilize the instantaneous physical/link layer performance report. For video and audio applications, the user data can tolerate certain amount of channel errors. Therefore, the packets, error-free or corrupted, should all be forwarded to the application, so that the media decoder has the right to decide whether to use or discard the packet. Indeed, most media decoders existing today support this feature. Furthermore, certain error indications like the locations of corrupted frames can guarantee perfect error detection and quick error recovery. When channel coding is applied to the application packets, this indication yields additional coding gain.
- CUDP outperforms the other two protocols simply due to the knowledge of the corrupted channel frame. However, as the congestion packet loss rate grows, which also contributes to the information loss, the advantage of CUDP shrinks.
- Using CUDP, the received video maintains good quality, even when a very significant fraction of video packets may have been dropped due to network congestion or received but corrupted by channel errors. For performance comparison, we have to take into account the impact of error concealment. Theoretically, even when FER is small, CUDP can effectively reduce the number of decoding failure. However, the information loss at those operating points can be recovered by error concealment so that the improvement in terms of video quality becomes less perceivable. Therefore, the advantage of CUDP also depends on the target video quality requirement as well as the operating point.

It should be pointed out that, although only MPEG video coding in used the simulations, the proposed system could be applied to other packetized video/audio/image, such as H.263, MPEG-4. In addition, throughout the paper, we assume that the delay requirement is satisfied by limiting or even precluding the retransmission at RLP layer and by CUDP. Future work includes investigating the actual performance profile taking into account of the delay due to retransmission at RLP layer and the packet loss due to real-time scheduling within wireless networks. In addition, we are working on utilizing BLAST architecture to further improve video transmission efficiency.

Acknowledgment

The authors would like to thank M. Sandell and H. Hwang for providing the simulation results for the baseline and BLAST wireless systems, and L. F. Wei for his useful comments on this work.

References

[1] K. Buchanan, R. Fudge, D. Mcfarlane, T. Phillps, A. Sasaki, and H. Xia, "IMT-2000: Service provider's perspective," IEEE personal communications, Aug. 1997.
[2] R. Pandya, D. Grillo, E. Lycksell, P. Meiybegue, H. Okinaka, and M. Yabusaki, "IMT-2000 standard: Network aspects," IEEE personal communication, Aug. 1997.
[3] J. Postel, "User datagram protocol," Request for Comments, RFC 768, ISI, Aug. 1980.
[4] W. Stevens, *TCP/IP Illustrated*. Reading, MA: Addison-Wesley, 1994, vol. 1.
[5] T. Bova and T. Krivoruchk. Reliable UDP protocol. Internet Draft, Network Working Group. [Online]. Available: draft-ietf-sigtran-reliable-udp-00.txt.
[6] L. Larzon, M. Degermark, and S. Pink, "Efficient use of wireless bandwidth for multimedia applications," in *Proc. MoMuc'99*, San Diego, CA, Nov. 1999, pp. 187–193.
[7] R. Blahut, *Theory and Practice of Error Control Codes*. Reading, MA: Addison-Wesley, 1983.
[8] D. Budge, R. McKenzie, W. Mills, and P. Long, "Media-independent error correction using RTP," Internet Engineering Task Force Internet Draft, May 1997.
[9] S. Wenger and G. Côté, "Using RFC2429 and H.263+ at low to medium bit-rates for low-latency applications," in *Packet Video'99*, New York, Apr. 1999.
[10] M. Gallant and F. Kossentini, "Robust and efficient layered H.263 internet video based on rate-distortion optimized joint source/channel coding," in *Packet Video'00*, Italy, 2000.
[11] S. Wenger and G. Côté, "Test model extension justification for internet/H.323 video transmission," Video Coding Experts Group, Document Q15-G-17, ITU Q15, Feb. 1999.
[12] J. Rosenberg and H. Schulzrinne. (1999, February) An RTP payload format for generic forward error correction. [Online]. Available: http://info.internet.isi.edu:80/in-drafts/files/draft-ietf-avt-fec-05.txt.
[13] "Medium Access Control (MAC) standard for cdma2000 spread spectrum systems,", TIA/EIA/IS-2000.3, Mar. 1999.
[14] A. Chockalingam and G. Bao, "Performance of TCP/RLP protocol stack on correlated fading DS-CDMA wireless links," *IEEE Trans. Veh. Technol.*, vol. 49, pp. 28–33, Jan. 2000.
[15] R. Van Nobelen, N. Seshadri, J. Whitehead, and S. Timiri, "An adaptive radio link protocol with enhanced data rates for GSM evolution," IEEE personal communication, Feb. 1999.
[16] "Point to point protocol,", Request for Comments, RFC 1661.
[17] H. Zheng and M. Lu, "A new RLP/PPP design for real-time services over wireless," Lucent Technologies, Patent application.
[18] J. Boyce, "Packet loss resilient transmission of MPEG video over the Internet," *Signal Process. Image Commun.*, pp. 7–24, Sept. 1999.
[19] G. J. Foschini, "Layered space-time architecture for wireless communication in a fading environment when using multiple antennas," *Bell Labs Tech. J.*, vol. 1, no. 2, pp. 41–59, Autumn 1996.
[20] Y. Wang and Q. F. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, 1998.

Haitao Zheng (S'98–A'99–M'00) received the B.S. degree in electrical engineering from Xian Jiaotong University, China, in 1995, and the M.S. and Ph.D. degrees in electrical engineering, from University of Maryland, College Park, 1998 and 1999, respectively.

From 1994 to 1995, she was with Video Technology Institute, Xian Jiaotong University. From 1995 to 1998, she was with the Institute for System Research Fellow, University of Maryland. Since August 1999, she has been with Wireless Communications Research Lab, Bell Labs, Lucent Technologies, Holmdel, NJ. Her research interests include signal processing for multimedia communications over wireless, Internet, and cable channels, where she has published extensively.

Dr. Zheng received the 1998–1999 George Harhalakis Outstanding Systems Engineering Graduate Student Award in recognition of outstanding contributions in cross-disciplinary research from the University of Maryland.

Jill Boyce (S'86–M'92) received the B.S. degree in electrical engineering from the University of Kansas, Lawrence, in 1988, and the M.S.E. degree in electrical engineering from Princeton University, Princeton, NJ, in 1990.

She is currently a Senior Member of the Technology Staff at Thomson Multimedia Corporate Research, Princeton, NJ. She was formerly with the Multimedia Communications Research Lab of Bell Laboratories, Lucent Technologies. Her research interests include video compression applications and algorithms, with an emphasis on maintaining quality of video transmitted over lossy channels, including IP and wireless networks. She holds 40 US patents.