Video Transport Over Ad Hoc Networks: Multistream Coding With Multipath Transport

Shiwen Mao, *Student Member, IEEE*, Shunan Lin, *Student Member, IEEE*, Shivendra S. Panwar, *Senior Member, IEEE*, Yao Wang, *Senior Member, IEEE*, and Emre Celebi, *Student Member, IEEE*

***Abstract*—Enabling video transport over ad hoc networks is more challenging than over other wireless networks. The wireless links in an ad hoc network are highly error prone and can go down frequently because of node mobility, interference, channel fading, and the lack of infrastructure. However, the mesh topology of ad hoc networks implies that it is possible to establish multiple paths between a source and a destination. Indeed, multipath transport provides an extra degree of freedom in designing error resilient video coding and transport schemes.**

In this paper, we propose to combine multistream coding with multipath transport, to show that, in addition to traditional error control techniques, path diversity provides an effective means to combat transmission error in ad hoc networks. The schemes that we have examined are: 1) feedback based reference picture selection; 2) layered coding with selective automatic repeat request; and 3) multiple description motion compensation coding. All these techniques are based on the motion compensated prediction technique found in modern video coding standards. We studied the performance of these three schemes via extensive simulations using both Markov channel models and OPNET Modeler. To further validate the viability and performance advantages of these schemes, we implemented an ad hoc multiple path video streaming testbed using notebook computers and IEEE 802.11b cards. The results show that great improvement in video quality can be achieved over the standard schemes with limited additional cost. Each of these three video coding/transport techniques is best suited for a particular environment, depending on the availability of a feedback channel, the end-to-end delay constraint, and the error characteristics of the paths.

***Index Terms*—Ad hoc networks, error resilience, IEEE 802.11, multipath transport, video transport, wireless networks.**

I. Introduction

AD HOC NETWORKS are multihop wireless networks without a preinstalled infrastructure. They can be deployed instantly in situations where infrastructure is unavailable (e.g., disaster recovery), or where infrastructure is difficult to install (e.g., battlefields). It is maturing as a means to provide ubiquitous untethered communication. With the increase both in the bandwidth of wireless channels and in the computing power of mobile devices, it is expected that video service will be offered over ad hoc networks in the near future.

Ad hoc networks pose a great challenge to video transport. There is no fixed infrastructure and the topology is frequently changing due to node mobility. Therefore, links are continuously established and broken. The availability and quality of a link further fluctuates due to channel fading and interference from other transmitting users. In addition, an end-to-end path consists of a number of wireless links. Thus, transmission loss in ad hoc networks is more frequent than that in wireless networks with single-hop wireless paths connecting nodes to the wireline infrastructure. The most popular media access control (MAC) scheme, the carrier sensing multiple access/collision avoidance (CSMA/CA) scheme [1], is designed for best-effort data. It provides no hard guarantees for a session's bandwidth and delay. Although bandwidth reservation is possible with MAC schemes based on time-division multiple-access (TDMA) or code-division multiple-access (CDMA), practical implementations of these schemes are nontrivial because of the synchronization or code assignment problems when node mobility is allowed [2].

Video transport typically requires stringent bandwidth and delay guarantees. However, it is very hard to maintain an end-to-end route, which is both stable and has enough bandwidth in an ad hoc network. Furthermore, compressed video is susceptible to transmission errors. For example, a single bit error often causes a loss of synchronization when variable length coding (VLC) is used. Moreover, the motion compensated prediction (MCP) technique is widely used in modern video coding standards. In MCP, a frame is first predicted from a previous coded frame (called reference picture) and then the prediction error is encoded and transmitted. Although MCP achieves high coding efficiency by exploiting the temporal correlation between adjacent frames, it makes the reconstruction of a frame depend on the successful reconstruction of its reference picture. Without effective error protection and concealment, a lost packet in a frame can cause not only error within this frame, but also errors in many following frames, even when all the following frames are correctly received [3].

Given the error-prone nature of ad hoc network paths and the susceptibility of compressed video to transmission errors, effective error control is needed. Traditional techniques, including forward error correction (FEC) and automatic repeat

Manuscript received October 15, 2002; revised May 1, 2003. This work was supported in part by the National Science Foundation under Grant ANI 0081375, in part by the New York State Center for Advanced Technology in Telecommunications (CATT) and the Wireless Internet Center for Advanced Technology (WICAT), Polytechnic University, Brooklyn, NY. This paper was presented in part at the IEEE International Conference on Multimedia and Expo (ICME), Tokyo, Japan, August 2001, and at the Vehicular Technology Conference, Atlantic City, NJ, October 2001.

S. Mao, S. Lin, S. S. Panwar, and Y. Wang are with the Department of Electrical and Computer Engineering, Polytechnic University, Brooklyn, NY 11373 USA (e-mail: smao@ieee.org; slin@vision.poly.edu; panwar@catt.poly.edu; yao@vision.poly.edu).

E. Celebi is with the Department of Computer and Information Science, Polytechnic University, Brooklyn, NY, 11373 USA (e-mail: celebi@ieee.org.).

Digital Object Identifier 10.1109/JSAC.2003.815965

request (ARQ), must be adapted to take into consideration the delay constraint and the error propagation problem [4]. In ad hoc networks, wireless links break down the traditional concept of topology, which is not constrained by physical cable connections anymore. Although user mobility makes links volatile, it provides variability of topology. On the one hand, a link may break when nodes move away from each other. On the other hand, it is possible to quickly find new routes formed in a new topology. Furthermore, the mesh topology of ad hoc networks implies the existence of multiple routes between two nodes. Given multiple paths, a video stream can be divided into multiple substreams and each substream is sent on one of the paths. If these paths are disjoint, the losses experienced by the substreams would be relatively independent. Therefore, better error resilience can be achieved when traffic dispersion is performed appropriately and with effective error control for the substreams. In a manner similar to multiantenna diversity that improves the capacity of wireless networks, path diversity can also be exploited to improve the capacity of ad hoc networks. Indeed, multipath transport (MPT) provides an extra degree of freedom in designing video coding and transport schemes.

In this paper, we propose three MCP-based video transport techniques for mobile ad hoc networks. These schemes take advantage of path diversity to achieve better performance. Compared with the coding methods considered in our previous work [5], these techniques can achieve significantly higher coding efficiency and are compliant with the H.26× and MPEG series standards (possibly with simple modifications). The techniques that we have examined include:

1) a feedback based reference picture selection scheme (RPS) [6];
2) a layered coding (LC) with selective ARQ scheme (LC with ARQ) [7];
3) a multiple description motion compensation coding scheme (MDMC) [8].

We studied the performance of these three schemes via a *top-down* approach. First ,we used a popular Markov link model [6], [9], where lower layer detail is embodied in the bursty errors generated. This simple model enables us to examine the system performance over a wide range of pack loss rates and loss patterns. Next, lower layer details, including user mobility, multipath routing, and the MAC layer are taken into account in the OPNET simulations [10], which provide a more realistic view of the impact of these factors on the system performance. Furthermore, we implemented an ad hoc video streaming testbed using notebook computers with IEEE 802.11b cards. This further validates the viability and performance advantages of these schemes. The results of our experiments show that video transport is viable in ad hoc networks given careful cross-layer design. Combining multistream coding with MPT improves video quality, as compared with traditional schemes where a single path is used. Each of these three techniques is best suited for a particular environment, depending on the availability of feedback channels, the end-to-end delay constraint, and the error characteristics of the paths.

The remainder of the paper is organized as follows. In Section II, we present the general architecture of multistream video coding with MPT. We also discuss related issues and prior work in this section. Next, the three multistream coding and MPT schemes are discussed in Section III. Sections IV and V present the performance study of these schemes using the Markov models and OPNET Modeler, respectively. Our experimental results with an ad hoc network video streaming testbed are reported in Section VI. Section VII provides discussion and conclusions.

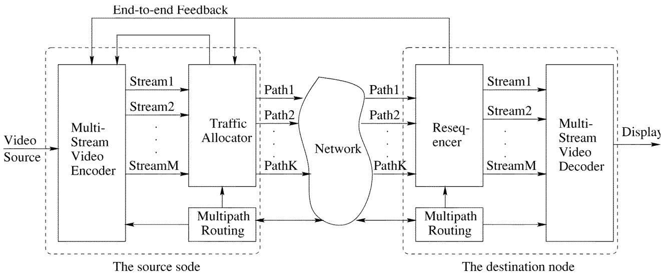


Fig. 1. General architecture of the proposed system using multistream coding and multipath transport.

II. Video Transport With Path Diversity

A. General Architecture

The general architecture of the proposed system is shown in Fig. 1. In the architecture, a multipath routing layer sets up K paths between the source and destination, each with a set of quality-of-service (QoS) parameters in terms of bandwidth, delay, and loss probabilities. The transport layer continuously monitors path QoS parameters and returns such information to the sender. Based on path quality information, the encoder generates M substreams. Packets from the substreams are dispersed by the traffic allocator among the K paths. At the receiver, packets arriving from all the paths are put into a resequencing buffer where they are reassembled into M substreams after a preset playout delay. Some or all the packets assigned to a path may be lost or overdue. Limited retransmission of lost packets may or may not be invoked, depending on the encoding scheme and the end-to-end delay constraint. The decoder will attempt to reconstruct a video sequence from the received substreams.

B. Related Issues and Challenges

A key to the success of the proposed system is the close interaction between the source coder and the transport layer, which entails careful cross-layer design. We will highlight this interaction in the following discussion.

1) Multistream Video Coding: For MPT to be helpful for sending compressed video, one must carefully design the video coder to generate substreams so that the loss in one substream does not adversely affect the decoding of other substreams. However, this relative independence between the substreams should not be obtained at a great expense of coding efficiency. Therefore, the multistream encoder should strive to achieve a good tradeoff between the coding efficiency and error resilience. In addition, one must consider what is feasible in terms of transport layer error control, when designing the source coder.

Obviously, one way to generate multiple substreams is to use a standard video codec and split the resulting bitstream into multiple substreams. An intelligent splitting scheme is needed to split the bit stream at the boundary of independently decodable units. Otherwise, a lost substream will make the received ones from other paths useless. A simple way to accomplish this is to send the frames to the paths in a round robin manner, e.g., all odd frames are sent to path 1 and all even frames are sent to path 2. In order to completely avoid the dependency between substreams, the frames sent on one path should be predictively coded with respect to the frames on the same path only. This method is in fact an option available in the $\mathrm{H.263+}$ standard [video redundancy coding (VRC)] [11]. However, compared with predicting a frame from its immediate neighbor, VRC requires significantly higher bit rates. Also, although this method can prevent the loss in one path from affecting frames in the other path, error propagation still exists within frames in the same path. In this paper, we introduce a feedback based reference picture selection method, which circumvents these two problems of VRC. This method is introduced in Section III-A.

Another natural way of generating multiple streams is by using *layered video coding*, which is very useful in coping with the heterogeneity of user access rates, in network link capacities, and in link reliability. A layered coder encodes video into several layers. The base layer (BL), which includes the crucial part of the video frames, guarantees a basic display quality. Each enhancement layer (EL) correctly received improves the video quality. But without the BL, video frames cannot be reconstructed sufficiently. Usually, EL packets may be dropped at a congested node to protect BL packets, and BL packets are better protected with FEC or ARQ [12]. When combined with MPT, it is desirable to transmit the BL substream on the best route. The source may sort the paths according to their loss characteristics, inferred from QoS feedback (e.g., receiver reports in the real-time transport protocol (RTP) [13]). Alternatively, the multipath routing layer may organize the route cache according to some performance metrics (number of hops, mean loss rate in the last time window, etc.). In this paper, we consider an approach that protects the BL by retransmitting lost BL packets on the path carrying the EL packets. This method is described in Section III-B.

Instead of generating substreams that are unequal in their importance, *multiple description coding* (MDC) generates multiple equally important streams, each giving a low but acceptable quality. A high-quality reconstruction is decodable from all bit streams together, while a lower, but still acceptable quality reconstruction is achievable if only one stream is received. The correlation among the substreams introduced at the encoder makes it possible to partially recover lost information of one substream, using information carried in other correctly received substreams. However, such a correlation limits the achievable coding efficiency, as compared with a conventional coder designed to maximize it. An excellent review of the theoretical bounds and proposed MDC algorithms can be found in [14]. In designing a MCP-based multiple description (MD) video codec, a key challenge is how to control the mismatch between the reference frames used in the encoder and those used in the decoder caused by transmission errors. Among several MD video coding schemes proposed so far [8], [11], [15], [16], we chose the MDMC method (Section III-C), because it outperformed other MD video coding methods in our previous studies [8]. With MDC, the transport layer design can be simpler than with layered coding. Because all the descriptions are equally important, the transport layer does not need to protect one stream more than another. Also, because each description alone can provide a low but acceptable quality, no retransmission is required, making MDC more suitable for applications with stringent delay requirements. Though the focus of this paper is on point-to-point communications, MDC can also be considered for multicast applications, where retransmissions are best avoided [17].

2) Multipath Transport: MPT has been studied in the past in wireline networks for: i) increased aggregate capacity; ii) better load balancing; and iii) path redundancy for failure recovery [18]–[20]. The research effort on MPT can be roughly divided into the following two categories.

a) Multipath routing, which focuses on finding multiple routes for a source-destination pair, and on how to select a maximally disjoint set of routes from the multiple routes found [21]–[24];

b) Traffic dispersion, which focuses on how to allocate traffic to multiple end-to-end routes [25], [26]. Generally, traffic dispersion can be performed with different granularities. [27] is an excellent survey of this topic.

The particular communication environment of wireless ad hoc networks makes MPT very appealing. In ad hoc networks: i) individual links may not have adequate capacity to support a high bandwidth service[1]; ii) a high loss rate is typical; and iii) links are unreliable. MPT can provide larger aggregate bandwidth and load balancing for ad hoc video applications. In addition, the path diversity inherent in MPT can provide better error resilience performance. Furthermore, many of the ad hoc network routing protocols, e.g., dynamic source routing (DSR) [28], ad hoc on-demand distance vector (AODV) [29], and zone routing protocol (ZRP) [30], are able to return multiple paths in response to a route query. Multipath routing can be implemented by extending these protocols with limited additional complexity.

There are many challenges in supporting MPT in ad hoc networks. First, from multiple paths returned by a route query, the routing process should select a set of maximally disjoint paths. Shared or nearby links of the paths could make the loss processes of the substreams correlated, which reduces the benefit of using MPT [31]. Algorithms for finding disjoint paths are presented in [21], [22]. Second, finding and maintaining multiple paths requires higher complexity and may cause additional overhead on traffic load (e.g., more route replies received). However, caching multiple routes to any destination allows prompt reaction to route changes. If a backup path is found in the cache,

[1] Although in some cases the nominal bandwidth of a wireless link is comparable to that of a wireline link, the available bandwidth may vary with signal strength as in IEEE 802.11b. In addition, capacity lost due to protocol overhead in ad hoc networks is much higher than that in wireline networks (e.g., RTS, CTS, ACK packets, and the 30-byte frame header in IEEE 802.11b).

there is no need to send new route queries. Rerouting delay and routing overhead may be reduced in this case. These problems should be addressed carefully in the design of a multipath transport protocol to balance its benefits. Third, a problem inherent in MPT is the additional delay and complexity in packet resequencing. Previous work shows that resequencing delay and buffer requirement are moderate if the traffic allocator in Fig. 1 is carefully designed [32], [33].

C. Related Work

Due to the availability of a variety of network access technologies, as well as the reduction in their costs, there is strong interest in taking advantage of multihomed hosts to get increased aggregate bandwidth and higher reliability. Proposals in the transport layer include [20], [34]–[36]. In [34], a protocol called meta-transmission control protocol (meta-TCP) maintaining multiple TCP connections for a session was designed for data transport. The stream control transmission protocol (SCTP) [20] was initially designed for reliable delivery of signaling messages in Internet protocol (IP) networks using path redundancy. There are now proposals to adapt it for data traffic in the Internet and in wireless networks [35], [36]. These papers focus on the higher aggregate usable bandwidth obtained and on how to perform TCP congestion control over multiple paths. Multiflow management can also be carried out at the application layer. In [5] and [37], an extension of the RTP [13], called meta-RTP, was proposed. Meta-RTP sits on top of RTP in the protocol stack, performing traffic allocation at the sender and resequencing at the receiver for real-time sessions.

Recently, several interesting proposals on delivering audio and video over Internet and wireless networks using multiple paths have been introduced. The study in [5] and [37] was, to the best of our knowledge, the first to investigate image and video transport using MPT in a multihop wireless radio network. Although it provided some very useful insights, the coders considered there treated individual frames of a video sequence independently, and consequently are not very efficient. There are several interesting papers on applying MPT for Internet multimedia streaming. In [38], MDC is combined with path diversity for video streaming in the Internet. A four-state model is proposed to capture the distortion behavior of a MD source. The problem of MD video downloading in content distribution networks (CDN) using a number of servers is studied in [39]. It is reported that 20% to 40% reductions in distortion can be achieved by using this *many-to-one* approach. Similarly, it is shown in [40] that using multiple senders and FEC in data downloading effectively reduces packet loss rates. An interesting study on realtime multistream voice communication through disjoint paths is given in [41], where multiple redundant descriptions of a voice stream are sent over paths provided by different Internet service providers. Both significant reductions in end-to-end latency and packet loss rate are observed. In recent work [42], the RPS scheme in [6] was extended by using rate-distortion optimized long memory reference picture selection, and using probes for path status prediction.

This paper differs from previous work discussed thus far in many aspects. First, we focus on video transport, as compared with general elastic data transport using TCP [20], [34]–[36]. We perform traffic partitioning in the application layer and use user datagram protocol (UDP) in the transport layer. We perform traffic dispersion on the substream level for the results shown in this paper.[2] Compared with meta-RTP [5], [37] our transport schemes require no integrity in each substream and are more flexible. Second, we study multipath video transport in ad hoc networks, while prior work focuses on Internet video streaming [38]–[42]. It is much more challenging to transport video in ad hoc networks than in a wireline network, e.g., the Internet, as discussed in Section I. Moreover, there is the well-known assumption that all packet losses in the Internet are caused by congestion [43], and MPT is mainly used in the Internet to alleviate congestion (i.e., load balancing). In an ad hoc network, in addition to congestion in mobile nodes, packets are also lost because the wireless links are unreliable. Therefore, the benefit of using MPT, in addition to load balancing, is error resilience through path diversity. Since the up and down status of the paths are relatively independent of each other, it is possible to apply efficient error control exploiting this feature to improve video quality. Third, the multistream video coding schemes we proposed are all MCP-based with high coding efficiency and are more compliant with modern video coding standards, as compared with [5], [37]. MDMC is a new multiple description video coding technique and [16] describes its algorithm and its performance using abstract channel models. In this paper, we study the performance of MDMC in ad hoc networks and performed extensive performance studies of MDMC under a more realistic network setting. Fourth, we extend the DSR protocol [28] to support multiple path routing. With our extension, multiple maximally disjoint routes are selected from all the routes returned by a route query, with only limited increase in the routing overhead. Fifth, for performance evaluation, we adopt a realistic model which includes all the layers except the physical layer using OPNET Modeler [10]. We believe this cross-layer model provides a realistic view for video transport over ad hoc networks.

In terms of implementation work, a number of ad hoc testbeds have been built recently [44], [45]. These mainly focus on the performance of ad hoc routing protocols, physical layer characteristics, scalability issues, and integration of ad hoc networks with the Internet for data transport. In [46], a firewall is inserted between the source and destination, which drops video packets according to a Markov channel model [9]. So far as we know, the testbed we developed is the first effort in combining multistream video coding and MPT for video transport in ad hoc networks.

III. Proposed Video Coding and Transport Schemes

One of the challenges when utilizing path diversity for video transmission is how to generate multiple coded substreams

[2] Finer packet-level traffic dispersion schemes can be supported by our schemes when more than two paths are available.

to feed the multiple paths. We consider three types of coding schemes that differ in terms of their requirements for the transport-layer support. These three schemes are all built on top of the block-based hybrid coding framework using MCP and discrete cosine transform (DCT), which is employed by all existing video coding standards. This way, the loss of coding efficiency is limited and the source codec can be implemented by introducing minimal modifications to codecs following existing standards. We present these three methods separately in the subsequent subsections, followed by comparison and discussion.

A. Feedback Based Reference Picture Selection

As discussed above, one of the main challenges in MCP-based video coding for ad hoc networks is how to limit the extent of error propagation caused by loss on a bad path, and yet minimize the loss in coding efficiency. As mentioned in Section II-B, one simple approach to generate two substreams is the VRC option in the $\mathrm{H}.263+$ standard [11], which codes the even and odd frames as two separate substreams and perform temporal prediction within each substream. However, compared with predicting a frame from its immediate neighbor, VRC requires significantly higher bit rates. Also, although this method can prevent the loss in one path from affecting frames in the other path, error propagation still exists within the same path. We note that there is no reason to forbid one path from using another path's frames as reference if all the paths are good. Motivated by this observation, we propose to choose the reference frames as follows: based on feedback and predicted path status, always choose the last frame that is believed to have been correctly received as the reference frame.

Specifically, we sent the coded frames on separate paths. The mapping of frames to paths depends on the available bandwidth on each path. For example, in the two-path case, if both paths have the same bandwidth, then even frames are sent on path 1, and odd frames on path 2. We assume that a feedback message is sent for each frame by the decoder. If any packet in a frame is lost, the decoder sends a negative feedback (NACK) for that frame. Otherwise, it sends a positive feedback (ACK). The feedback information for a frame may be sent on the same path as the frame, or on a different path. An encoder receives the feedback message for frame $n - \mathrm{RTT}$ when it is coding frame n, where round-trip time (RTT) is measured in frame intervals.

Furthermore, once a NACK is received for a frame delivered on one path, we assume that the path remains "bad" until an ACK is received. Similarly, we assume the path stays in the "good" status until a NACK is received. When encoding a new frame, the encoder deduces the last correctly decoded frame, based on the feedback messages received up to this time, and uses that frame as the reference frame. This scheme works well when the loss of a path is bursty, which is typical in ad hoc networks. A more sophisticated scheme may adopt a threshold for the NACKs and ACKs for a given window of time and switch the reference frame only when the threshold is exceeded.

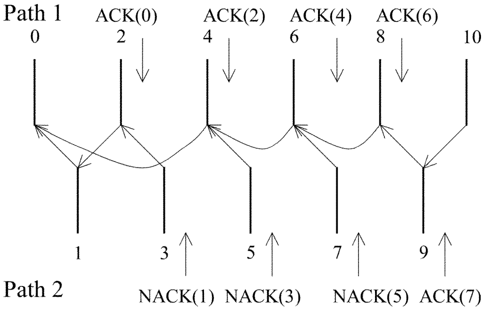


Fig. 2. Illustration of the RPS scheme. The arrow associated with each frame indicates the reference used in coding that frame.

Fig. 2 is an example of the proposed RPS scheme where $\mathrm{RTT} = 3$. When NACK(1) is received at the time when frame 4 is being encoded, the encoder knows that frames 2 and 3 cannot be decoded correctly due to error propagation. Therefore, frame 0 is chosen as the reference for frame 4 and path 2 is set to "bad" status. When frame 6 is coded, the encoder uses frame 4 instead of frame 5 as reference frame, because path 2 is still in the "bad" status. When ACK(7) is received, path 2 is changed to "good" status. Frame 9 is then chosen as the reference of frame 10.

The RPS scheme offers a good tradeoff between coding efficiency and error resilience. When both paths are good, RPS uses the immediate neighboring frame as reference, thereby achieving the highest possible prediction gain and coding efficiency. When one path is bad, the encoder avoids using any frames that are affected by path errors, thereby minimizing the error propagation period. RPS has higher coding efficiency than the schemes using fixed prediction distances [11] and error propagation in each substream is effectively suppressed. These improvements are achieved by using a frame buffer to store several previous coded frames as possible references, and by using feedback for path status prediction. Note that in this scheme feedback is used to control the operation of the encoder. No retransmission is invoked.

B. Layered Coding With Selective ARQ

This is a scheme using layered video coding. With this scheme, a raw video stream is coded into two layers, a BL and an EL. We follow the SNR profile in the H.263 standard [48] when generating the layers. A BL frame is encoded using the standard predictive video coding technique. Note that because the BL coding uses only the previous BL picture for prediction, this coding method has a lower coding efficiency than a standard single layer coder. This loss in coding efficiency is, however, justified by increased error resilience: a lost EL packet will not affect the BL pictures. Good quality is, thus, guaranteed if the BL packets are delivered error-free or at a very low loss rate. There are three prediction options in the H.263 standard for enhancement layer coding: UPWARD prediction, in which the BL reconstruction of current frame is used as the prediction of the enhancement layer, FORWARD prediction, in which the enhancement layer reconstruction from the previous frame is used as the prediction, and BIDIRECTION prediction, in which the average of BL reconstruction and enhancement layer reconstruction is used. The LC with ARQ codec selects from the three prediction options the one that has the best coding gain. Although this approach is optimal in terms of coding efficiency for the enhancement layer, error propagation can still occur in the EL pictures.

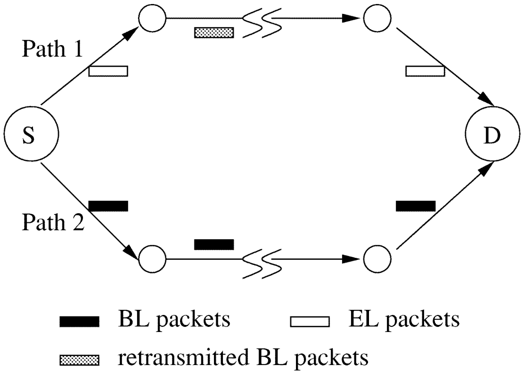


Fig. 3. Two-path layered video transmission model with end-to-end ARQ for BL packets.

Given two paths, the traffic allocator sends the BL packets on one path (the better path in terms of loss probability when the two paths are asymmetric) and the EL packets on the other path. The receiver sends selective ARQ requests to the sender to report BL packet losses. To increase the reliability of the feedback, a copy of the ARQ request is sent on both paths. When the sender receives an ARQ request, it retransmits the requested BL packet on the EL path, as illustrated in Fig. 3. The transmission bit rate for the EL may vary with the bit rate spent on BL retransmission. For video streaming applications where video is typically pre-encoded off-line, a simple rate control method is used for the EL path: when a BL packet is retransmitted on the EL path, one or more EL packets are dropped to satisfy the target transmission rate on the EL path.

Our observations show that a multiple hop wireless path behaves more in an on–off fashion with bursty packet losses. If there is a BL packet loss, the BL route is most likely to be broken. Moreover, if the loss is caused by congestion at an intermediate node, using the BL path for retransmission may make the congestion more severe. If disjoint paths are used, path diversity implies that when the BL path is down or congested, it is less likely that the EL path is also down or congested. Therefore, retransmission using the EL path is likely to have a higher success probability and lower delay.

As discussed in Section II-B, we assume either the sender continuously estimates the states of the paths based on received ARQ requests or QoS reports, or the multipath routing process orders the paths according to their loss characteristics. In the first case, a burst of ARQ requests received at the sender implies that the BL path is in a "bad" state. If the inferred EL path state is better, the sender may switch the paths. In the latter case, an intermediate node may send an error report back to the source after it drops a packet (e.g., an error report in DSR [28], or an Internet control message protocol (ICMP) unreachable error message [47]). These error reports will trigger the routing process to reorder the paths or initiate a new route query.

C. Multiple Description Motion Compensation

Unlike the above two techniques, MDMC is a multiple description coding scheme which does not depend on the availability of feedback channels. Because paths in ad hoc networks change between "up" and "down" state very often, each description experiences bursty packet losses. Therefore, we employ the *packet-loss mode* of MDMC presented in [10]. It uses a linear superposition of two predictions from two previously coded frames. In the MDMC encoder, the central prediction is obtained by

TABLE I
COMPARISON OF THE THREE SCHEMES

-	RPS	LC with ARQ	MDMC
Feedback Needed	Yes	Yes	No
Decoding Delay	No	$\geq RTT$	No
Redundancy Controlled by	Error rates and RTT	Error rates and BL quality	Encoding parameters
Additional Buffer	$\geq RTT$ frames	$\geq RTT$ frames	1 frame

$$\hat{\psi}(n) = a_1\tilde{\psi}_e(n-1) + (1-a_1)\tilde{\psi}_e(n-2) \tag{1}$$

where $\tilde{\psi}_e(n-1)$ and $\tilde{\psi}_e(n-2)$ are motion compensated predicted signals constructed from two previously encoded frames $\psi_e(n-1)$ and $\psi_e(n-2)$, respectively. The central prediction error $e_0(n) = \psi(n) - \hat{\psi}(n)$ is quantized by quantizer $Q_0(\cdot)$ to $\tilde{e_0}(n)$. The quantized prediction error and motion vectors for even frames are sent on one path, and those for odd frames are sent on another path. In the decoder, if frame $n-1$ is received, frame n is reconstructed using

$$\psi_d(n) = a_1\tilde{\psi}_d(n-1) + (1-a_1)\tilde{\psi}_d(n-2) + \tilde{e}_0(n) \tag{2}$$

where $\tilde{\psi}_d(n)$ represents motion compensated prediction from decoded frame n.

If frame $n-1$ is damaged but frame $n-2$ is received, the decoder only uses the reconstructed frame $n-2$ for prediction. To circumvent the mismatch between the predicted frames used in the encoder and the decoder, the signal $e_1(n) = \tilde{\psi}_e(n-2) - a_1\tilde{\psi}_e(n-1) - (1-a_1)\tilde{\psi}_e(n-2) - \tilde{e}_0(n)$ is quantized by another quantizer $Q_1(\cdot)$, which is typically coarser than $Q_0(\cdot)$, and the output $\tilde{e}_1(n)$ is sent along with other information on frame n. Now, when frame $n-1$ is damaged, the side decoder reconstructs frame n using

$$\psi_d(n) = \tilde{\psi}_d(n-2) + \tilde{e}_0(n) + \tilde{e}_1(n). \tag{3}$$

In addition, the lost frame $\psi(n-1)$ is estimated using

$$\tilde{\psi}_d(n-1) = \frac{1}{a_1}\Big(\psi_d(n) - (1-a_1)\tilde{\psi}_d(n-2) - \tilde{e}_0(n)\Big). \tag{4}$$

The MDMC codec offers a tradeoff between redundancy and distortion over a wide range by varying the coder parameters (the predictor coefficient a_1 and the quantization parameter of $Q_1(.)$). The efficiency of a MDMC codec depends on the selection of the parameters, which in turn depends on the estimation of the channel's error characteristics. There is only one additional buffer needed in MDMC compared with conventional codecs that use only one previous frame for prediction.

D. Comparison and Discussion

The three schemes have their respective advantages and disadvantages. Depending on the availability of a feedback channel, the delay constraint, and the error characteristics of the established paths, one technique may be better suited for an application than another. A comparison of these three schemes is given in Table I.

RPS is applicable when feedback channels are available. The redundancy depends on the distance between a current frame and its reference frame, which in turn depends on the packet loss rate and the RTT. When the paths are error-free, RPS has the highest encoding efficiency. Compared with ARQ-based schemes, there is no decoding delay incurred but additional buffers are still needed.

LC with ARQ is suitable when feedback channels are available and the application is such that the latency caused by retransmission is tolerable. The redundancy of this scheme comes from the fact that a frame is predicted from the BL reconstruction of the reference frame. It is difficult to control the amount of the redundancy introduced, which is more than the amount of redundancy introduced in the MDMC coder, when operating under the chosen set of parameters. This is why the LC approach has the lowest quality when packet loss rate is low. However, when the packet loss rate is high, this method can usually deliver the BL successfully, thus providing better video quality than the other two proposed schemes, at the cost of extra delay. The additional delay is at least RTT.

MDMC, unlike the other two, does not need feedback, nor does it incur additional decoding delay. It is easier to control the redundancy in MDMC by changing the predictors and the side quantizer. The redundancy can be achieved in a wider range than the above two schemes (even though the parameters of the MDMC coder are fixed in all the simulation studies reported here). Since MDMC needs no feedback, the video can be pre-encoded, which is desirable for video streaming applications. Note that this is not possible with the RPS scheme. The challenge with MDMC is how to adapt the coding parameters based on the error characteristics of the paths so that the added redundancy is appropriate.

IV. Performance Study Using Markov Models

In this section, we report on performance studies of the proposed schemes. The challenge is that the problem requires cross-layer treatment with a large set of parameters. To simplify the problem and focus on the key issues, we first study the performance of the schemes using Markov link models. The lower layer details are hidden and their impact on the video transport is embodied in the bursty errors generated by the Markov models. We will examine the impact of lower layer components such as user mobility, multipath routing, and the MAC layer using OPNET simulations in Section V.

A. Video Codec Implementations and Parameters

We implemented in software the proposed three video coding schemes on top of the public domain H.263+ codec [49]. For RPS, we added our reference picture selection algorithm by extending the RPS option provided by the standard. For LC with ARQ, we added a simple rate control algorithm for EL, as explained in Section III-B. For MDMC, the codec was modified to produce both central and side predictions in the INTER mode, and encodes central and side prediction errors using quantization parameters QP_0 and QP_1, respectively. More details about MDMC can be found in [8].

Error concealment is performed in the decoders when packets are lost. In the RPS decoder, the conventional *copy-from-previous-frame* method is used. In the LC decoder, if the BL is lost, the *copy-from-previous-frame* method is used. If the EL is lost but the BL is received, the frame is reconstructed using the BL only. In the MDMC decoder, the lost information can be recovered partially from the other description received (see Section III-C and [8]).

We use the quarter common intermediate format (QCIF), [176×144 Y pixels/frame, 88×72 Cb/Cr pixels/frame] sequence "Foreman" (first 200 frames from the original 30 fps sequence) encoded at 10 fps in the performance study of the schemes. The encoder generates two substreams with a bit rate of 59 kb/s each. The TMN8.0 [49] rate control method is used in RPS and LC with ARQ, but the frame layer rate control is disabled. In the simulations using Markov models, RTT is assumed to be three frame intervals. In MDMC, a_1 is set to 0.9, and the quantization parameter (QP_0, QP_1) is fixed at (8,15) [8], which achieves approximately the same bit rate as the other two schemes. The buffer size of RPS is set to 12 frames. If the selected reference frame is not found in the buffer, the nearest frame is used instead. In all the methods, 5% macroblock level intra-refreshments are used, which has been found to be effective in suppressing error propagation for the range of the packet loss rates considered. Each group of blocks (GOB) is packetized into a single packet, to make each packet independently decodable. In LC with ARQ transmission, the BL is transmitted on the better channel if the two channels are asymmetric. In the following simulations, we only allow a lost BL packet to be retransmitted once.

B. Modeling of Ad Hoc Routes Using Markov Models

In [31], replied routes for a route query are first broken into a pool of links and disjoint routes are assembled from the links in the pool. Motivated by this work, we model a multiple hop wireless route using the concatenation of a number of links, drawn randomly from a link pool and each is modeled by a Markov chain [9].

This model has many advantages. The link pool can be easily built using measurement data of ad hoc links. Furthermore, the loss pattern and loss rate are easily controlled. More complex packet loss processes can also be modeled using semi-Markov models. Performance analysis of the video codecs under a *full spectrum* of loss processes is possible with this technique. The disadvantage of this model is that it is a high level model. Details such as bandwidth and end-to-end delay variation, interference, and user mobility cannot be modeled accurately. These issues will be addressed in the OPNET models in Section IV-C.

C. Simulation Results Using Markov Channel Models

A three-state Markov model was used for each link with the states representing a "good," "bad" or "down" status for the link. The "down" state means the link is totally unavailable. The "good" state has a lower packet loss rate than the "bad" state. The packet loss rates we used are $p_0 = 1.0$, $p_1 \in [0.1\%, 20\%]$, and $p_2 = 0$, for the "down," the "bad," and the "good" states, respectively. The transition parameters are chosen to generate loss traces with desired loss rates and mean burst lengths. In

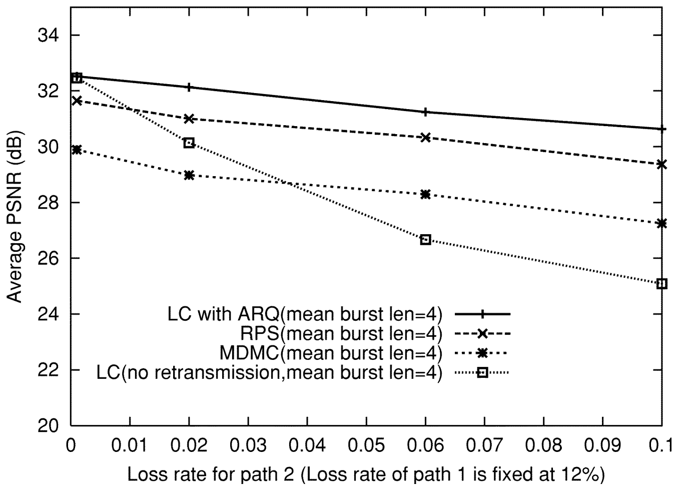


Fig. 4. Average PSNRs of the three schemes with asymmetric paths: Path 1's loss rate is fixed at 12% and path 2's loss rate varies from 0.1% to 10%.

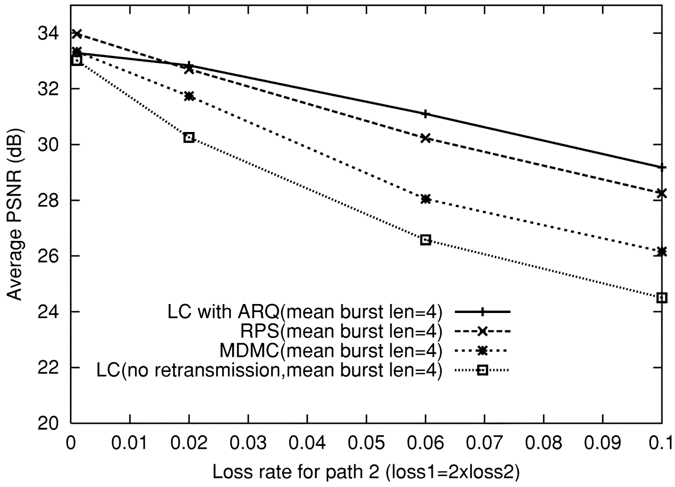


Fig. 5. Average PSNRs of the three schemes with asymmetric paths: Path 1's loss rate is twice of that of path 2.

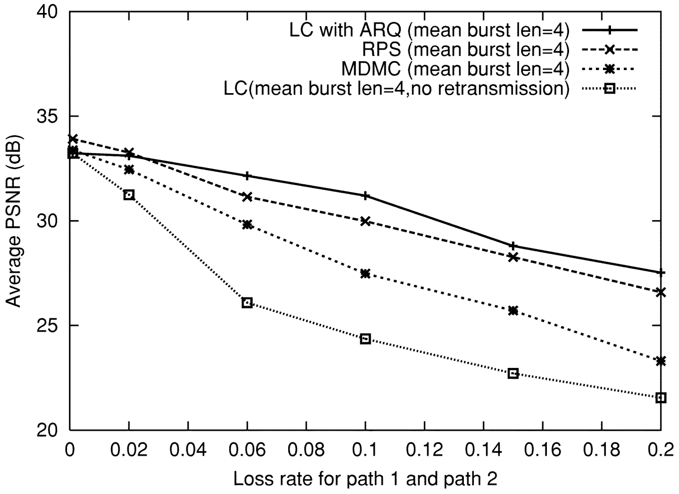


Fig. 6. Average PSNRs of the three schemes with symmetric paths: The mean burst length is fixed at 4 packets, while the loss rates varies.

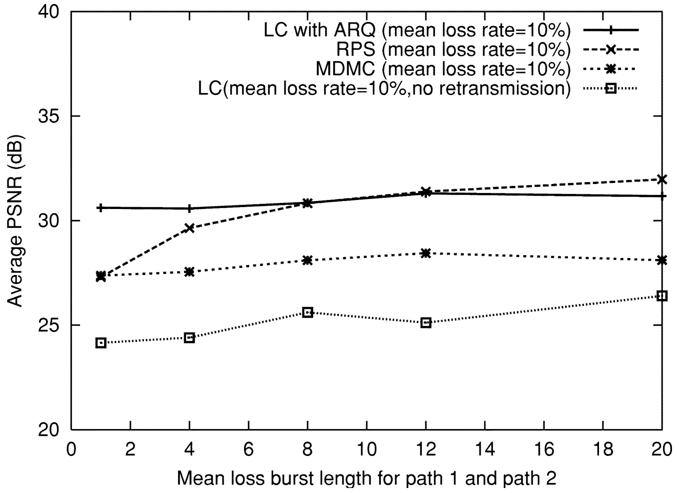


Fig. 7. Average PSNRs of the three schemes with symmetric paths: The loss rates are fixed at 10%, while the mean burst length varies.

our simulation, two paths were set up for each connection, and each path was continuously updated as follows: After every 2 s, two links were chosen randomly from a link pool to construct a new path. For the results reported in Figs. 4–7, the paths used are disjoint to each other. A video packet can go through a path correctly only when it goes through every link successfully. For each case studied in the following, a 400s video sequence is used by repeatedly concatenating the same video sequence 20 times. The error-free average peak-signal-to-noise ratio (PSNR) of the received video frames are 34.14, 33.31, and 33.47 dB for RPS, LC with ARQ, and MDMC, respectively.

The average PSNRs of decoded video sequences under various packet loss rates are given in Figs. 4–7. From the figures, we can conclude that the best choice depends on the channel characteristics, including error rate and error pattern (burst length), the application requirements, including delay constraint, and the differences among those channel characteristics (for example, symmetric or asymmetric). Specifically, the following observations can be made from the figures.

1) Effect of Packet Loss Rates: Generally, when the burst length is not higher than 9 packets, LC with ARQ has the best performances when the error rate is medium to high. A large burst of error may make the ARQ scheme less effective (see Fig. 7). RPS has the best performance at very low error rate due to its coding efficiency.

2) Effect of Path Symmetry: The paths may be symmetric or asymmetric in terms of packet loss rates. Comparing Figs. 5 and 6, when the average loss rate on the two paths is equal, LC without ARQ works better in asymmetric channels, while the performance of RPS, LC with ARQ, and MDMC is similar with either asymmetric or symmetric channels. For example, for the (2%, 4%) point in Fig. 5, LC with ARQ has average PSNR of 32.84 dB, while for the (3%, 3%) point in Fig. 6, LC with ARQ has an average PSNR of 32.87 dB. This broadly holds for all other points in both figures. With retransmission of lost BL packets on the EL path, the BL path does not need to be significantly better than the EL path. The fact that the video quality is insensitive to path symmetry with all three schemes is a blessing: This means that one does not need to provide special provisioning in the network to provide at least one high quality path.

3) Effect of Error Burst Lengths: From Fig. 7, we observe that in the low to intermediate burst length range considered, the performances of all the three schemes improves gradually as the burst length increases.[3] The reason is when the average packet loss rates are the same, a shorter burst length means more frames have errors, which causes more distortions at the video decoders; while a larger burst length means less frames have errors, which could be remedied with effective error concealment. When the mean error burst length increases, the RPS scheme gains most, since its path status prediction method works better with longer bursts. Note that this trend is true only up to a certain point. When the burst length increases beyond this point, so that all packets in two or more consecutive frames are corrupted by an error burst, the PSNR of decoded video will start to drop sharply.

4) Effect of Delay Constraint: When retransmission is allowed by the end-to-end delay constraint, LC with ARQ has the best performance among these three schemes, even only one retransmission is allowed. We also show in Fig. 4–7 the results for the LC scheme without retransmissions. This test is used to emulate the case when retransmission is not possible, because the delay constraint of the underlying application does not allow to do so, because it is not feasible to set up a feedback channel, or because end-to-end retransmission is not practical (e.g., video multicast). Without the ARQ protection of BL, the performance of LC is, as expected, the poorest among all the schemes. It shows that LC is effective only when the transport layer can provide efficient error control (e.g., FEC or ARQ) for the BL packets. Although LC with ARQ has the highest PSNR in the cases we studied, it is the most susceptible to a large end-to-end delay and imperfect feedback channels.

Note that the performance of the three schemes is also influenced by several other factors. For example, the performance of RPS varies with the RTT as well: the shorter the RTT is, the better RPS works. This is because when the RTT is shorter, the RPS encoder will be notified of the corrupted frames earlier and then it can stop error propagation earlier. Because MDMC does not require the set up of a feedback channel, it can be used for a wider range of applications.

V. Performance Study Using OPNET Models

The simplicity of the Markov model enables us to examine the performance of the proposed schemes over a wide range of packet loss patterns. In this section, we use the OPNET models to examine the impact of lower layer factors, which are not revealed by the Markov model. Using MDMC as an example, we show how multipath routing, MAC operation and user mobility affect video transport.

A. Multipath Routing Using Dynamic Source Routing

DSR is a source routing protocol proposed for mobile ad hoc networks, where intermediate nodes do not need to maintain up-to-date routing information for forwarding a transit packet since the packet carries the end-to-end path in its header. It is an on-demand protocol where route discovery is performed for a node only when there is data to be sent to that node [28].

```
W=getRouteFromReply(Route Reply);
if( (R0 is empty) and (R1 is empty) ) {
  copy W to RAND(R0,R1);
}
else if ( (R0 is not empty) and (R1 is empty) ) {
  copy W to R1;
}
else if ( (R0 is empty) and (R1 is not empty) ) {
  copy W to R0;
}
else {
  if(R0,R1,W have the same number of common nodes) {
    (R0,R1)=getTheTwoShortestRoutes(R0,R1,W);
  }
  else {
    (R0,R1)=getTheTwoMostDisjointRoutes(R0,R1,W);
  }
}
NOTE: W is the returned route, R0 and R1 are cached routes.
      RAND(R0,R1) returns R0 or R1 with equal probability
```

Fig. 8. Route updating algorithm.

There are a number of extensions of DSR to multipath routing. In [22], intermediate nodes are forbidden to reply to route queries. The destination node receives several copies of the same route query (each traverses a possibly different path) within a time window. At the end of the time window, the path with the shortest delay and another path which is most disjoint with the shortest path are returned to the source. In [50], in addition to a shortest path, backup paths from the source node and from each intermediate node of the shortest path are found to reduce the frequency of route request flooding.

We extended DSR to multipath DSR (MDSR) in the following way. Each node maintains two routes to a destination. We allow both the destination node and intermediate nodes to reply to a route query. When the destination node replies, it also copies the existing routes from its own route cache into the route reply, in addition to the route that the route query traversed. Another difference with the single path DSR is that when a node overhears a reply which carries a shorter route than the reply it plans to send, it still send its reply to the requesting source. This increases the number of different routes returned, giving the source a better choice from which to select two maximally disjoint paths from these replies. The first returned route is used by the originating packet. Then, the route cache is further updated (or optimized) as new replies for the same query arrive, using the algorithm in Fig. 8. This algorithm is a greedy algorithm in the sense that it always finds the best paths returned *so far*. Since with DSR, a reply with shorter route usually arrives earlier [28], this heuristic algorithm gives good performance without using a time window. As compared with [22], the routing delay with MDSR is smaller.

Although MDSR is capable of maintaining more than two routes, we only experimented with the two-path version, since the results in [50] indicate that the largest improvement is achieved by going from one to two or three paths. The MDSR models is built based on the OPNET DSR model [51].

[3] The burst length in Figs. 4–7 is measured in packets. With the QCIF video, there are nine packets per frame.

B. OPNET Simulation Setting

Using the OPNET model, we simulate an ad hoc network with 16 nodes in a 600 m×600 m region. Given the dimensions of the region, 16 nodes result in a density that maintains a connected network for most of the time [52]. Each node is randomly placed in the region initially. We used a version of the popular *random waypoint* mobility model, where each node first chooses a random destination in the region, then moves toward it at a *constant* speed. When it reaches the destination, it pauses for a constant time interval, chooses another destination randomly, and then moves toward the new destination [53]. Note that this is a simplified version of the Random Waypoint model. Since there is no randomness in the nodal speed, the convergence problem reported in [54] does not present itself here. We used a pause time of 1.0 s for all the experiments reported in this paper. The speed of the nodes varies from 0 m/s to 10 m/s, which models movement of pedestrians or vehicles in city streets.

We use the IEEE 802.11 protocol in the MAC layer working in the DCF mode. Its physical layer features, e.g., frequency hopping (FH), are not modeled. The channel has a bandwidth of 1 Mb/s. The transmission range is 250 m. If the sender of a packet is within this range of the receiver, and the sender has successfully accessed the channel during the transmission period, the packet is regarded as correctly received. The maximum number of link layer retransmissions is seven, after which the packet is dropped. UDP is used in the transport layer. We also implemented part of the RTP functionality [13], such as timestamping and sequence numbering in our model.

Among the 16 nodes, one is randomly chosen as the video source and another node is chosen as the video sink, where a 5 s playout buffer is used to absorb the jitter in received packets. The video source starts a session using two routes, sending two substreams of encoded video (59 kb/s each) to the sink. All other nodes generate background traffic to send to a randomly chosen destination. The interarrival time of the background packets is exponentially distributed with a mean of 0.2 s. The background packet has a constant length of 512 bits.

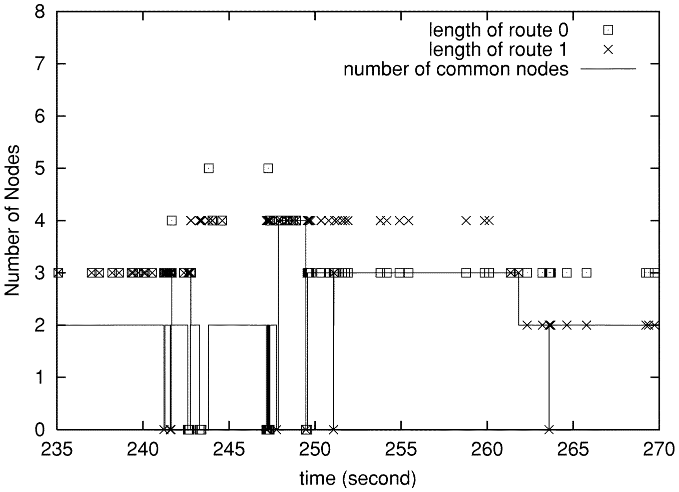


Fig. 9. Simulation results of 16 nodes moving in a 600 m × 600 m region at a speed of 10 m/s. Plotted are the traces of two routes to the video sink maintained by the video source during the simulation period.

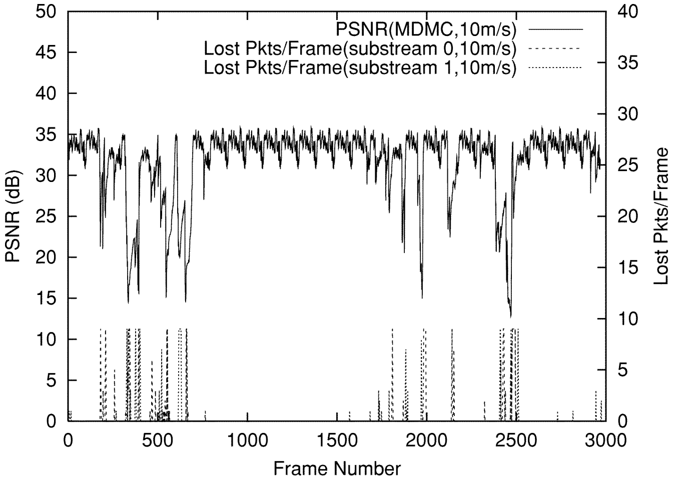


Fig. 10. PSNRs of the received frames with a MDMC codec using two paths. 16 nodes move in a 600 m × 600 m region at a speed of 10 m/s. Plotted on the right y axis are the lost packets per frame. The MSDR algorithm is used for route updates. The measured average loss rates of the two substreams are: (3.0%, 3.1%).

C. Simulation Results Using OPNET Models

1) MDSR Performance: First, we examined the performance of MDSR. Fig. 9 plots the traces of two routes maintained by the video source node during a simulation in which each node has a speed of 10 m/s. The length of a route is denoted by the total number of nodes the route traverses, including the source and the destination. Each point in the figure also means a route update: either a better route is found, or a route in use is broken. It can be seen that the routes are unstable. However, since the nodes are moving around rapidly, new neighbors are found and a new route is discovered shortly after the old route is down. This shows that mobility is both harmful and helpful. The lengths of the routes vary from 0 to 6 (5 hops, a little higher than the diameter of the network) during the simulation. For most of the simulation period, the route length is 2, which means direct communication between the source and the destination, or 3, which means a relay node is used in between. We also plot the number of common nodes between two paths. During most of the simulation period, this number is two, which means the two routes are disjoint except for the common source and destination.

Fig. 9 shows the period when the routes are the longest and most correlated. At the beginning of the period, the two paths are two hops each and are disjoint. Then, they get longer and have more common nodes, probably indicating that they are moving away from each other. After the 260th second, they become two hops again after shorter new paths are found. After the 262th second, the routes become disjoint again.

2) MPT Versus SPT: Next, we compare the performance of MPT with single path transport (SPT) in Figs. 10 and 11, where the same multistream video coder MDMC is used. For SPT, we used the NIST DSR model [51], which maintains a single path to a destination, while for MPT we used the MDSR model which is a multipath routing extension of [51]. We transmit both substreams on the same path in the SPT simulations. To alleviate the impact of bursty errors, we interleave the packets of the two descriptions with a two-frame interleaving interval. Description 1

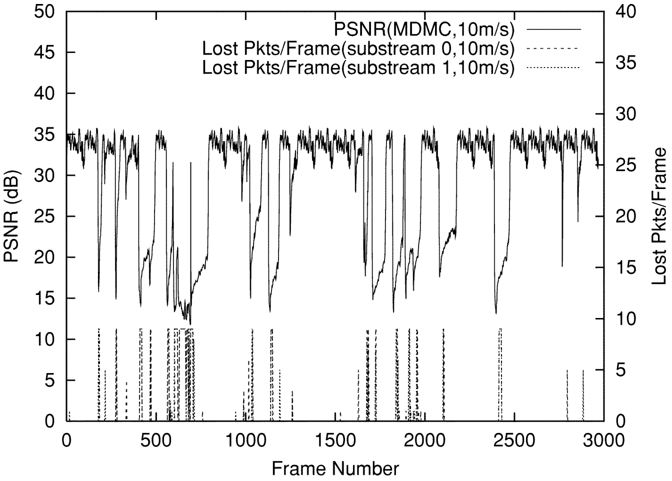
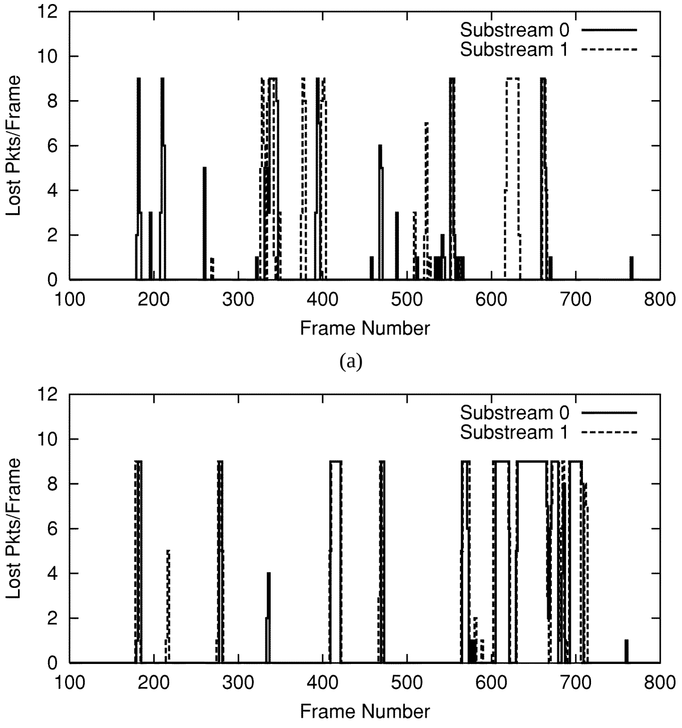


Fig. 11. PSNRs of the received frames with a MDMC codec using a single path. 16 nodes move in a 600 m × 600 m region at a speed of 10 m/s. Plotted on the right y axis are the lost packets per frame. The path is updated using the the NIST DSR model and both substreams are sent on the path using an interleaving interval of two frames. The measured average loss rates of the two substreams are: (6.3%,6.4%).

(2) packets for two even (odd) frames is followed by description 2 (1) packets for two odd (even) frames. We perform this experiment for the 16-node network, where each nodes moves at a speed of 10 m/s. The PSNR traces (using the left y axis) and loss traces (using the right y axis) using MDSR and DSR are plotted in Figs. 10 and 11, respectively. It can be seen that PSNR drops when there is loss in either substream. Also, the deepest drop occurs when a large burst of loss of one substream overlaps with that of the other substream. SPT has higher loss rates than MPT and, therefore, the PSNR curve in Fig. 11 has more frequent and severe drops than that in Fig. 10. It is obvious that SPT has poorer performance than MPT.

This is further illustrated in Fig. 12, which displays the zoomed versions of the loss traces of both simulations. The SPT traces [Fig. 12(b)] have more frequent packet losses than the MPT traces [Fig. 12(a)]. Even worse is that the packet losses of the substreams are strongly correlated in the SPT case. This has the most negative impact on the MDMC performance. Although we interleaved the substreams before transmitting, the burst length is too long, rendering the interleaving ineffective. A larger interleaving interval may help but at the cost of a larger end-to-end delay.

3) Impact of Mobility: We also examined the impact of mobility on video transport using MDMC as an example. Fig. 13 shows the PSNRs of the received video frames when the nodes are stationary. The PSNR curve in Fig. 13 is very stable, with only a few narrow drops. From the error trace below, we can see that the losses are mostly random, i.e., the error burst length is one packet for most of the time. When nodes begin to move around at a speed of 10 m/s, the PSNR curve in Fig. 10 is much worse with many more drops. The largest drops occur at the 500th, 2000th, and 2500th frames.[4] We conjecture that during these periods the source node and destination node were either far away from each other, or were in a hot-spot. We can see that in both figures, the valleys of the PSNR curve match the loss bursts drawn below. In Fig. 10, the loss bursts in this plot match the two routes' longest and most correlated periods in Fig. 9. Mobility clearly has a negative effect on video transport.



(a)

(b)

Fig. 12. Zoomed lost packet per frame traces of two substreams. (a) MPT case (corresponding to Fig. 10). (b) SPT case (corresponding to Fig. 11).

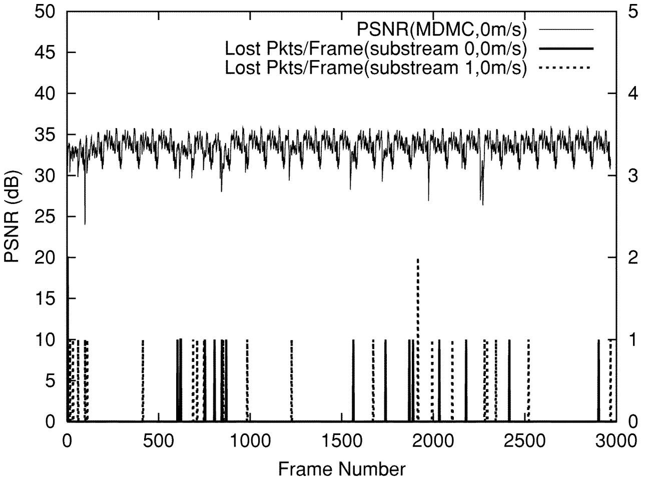
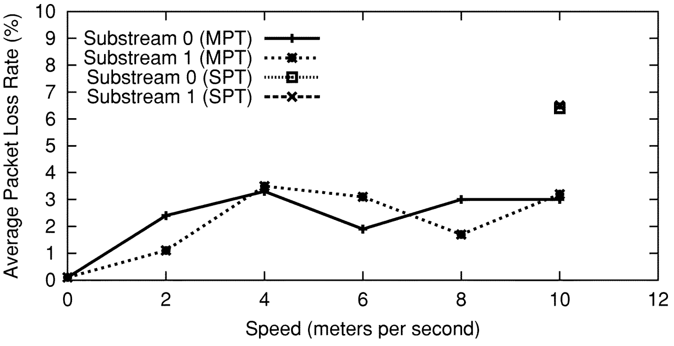


Fig. 13. PSNRs of the received frames with a MDMC codec. 16 nodes in a 600 m × 600 m region. Plotted on the right y axis are the lost packets per frame. The nodes are stationary.

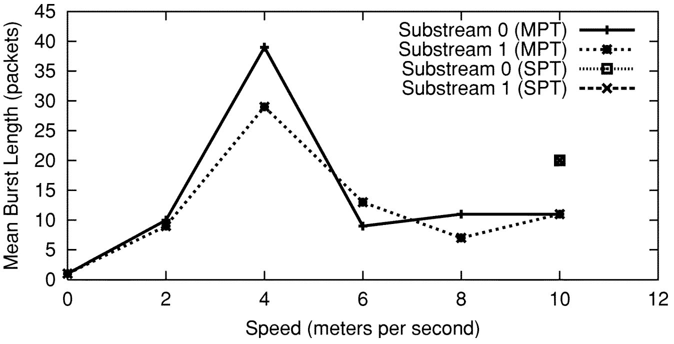
Fig. 14 shows the mean packet loss rates and the mean packet loss burst lengths of the two substreams during the simulations when the mobile speed varies from 0 m/s to 10 m/s. Fig. 15 is the resulting average PSNR for different speeds using the MDMC codec. There are several interesting observations.

1) The two routes maintained by MDSR are relatively symmetric in their error characteristics. Recall from Fig. 8

[4] Note that the frame rate is 10 frames/s. The frame number corresponding to the 250th second in Fig. 9 is 2500 in Fig. 10.



(a)



(b)

Fig. 14. Loss characteristics versus mobile speed for both MPT and SPT OPNET simlations. (a) Average packet loss rate. (b) Average error burst length.

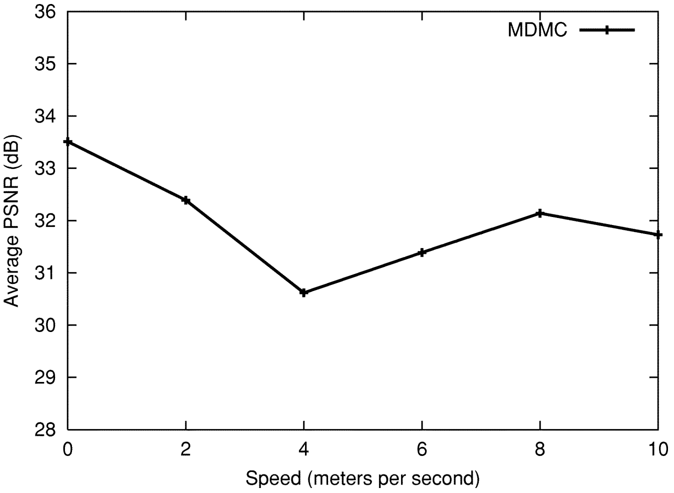


Fig. 15. Average PSNRversus node speed for the MDMC scheme from the OPNET simulations.

that the algorithm does not order the two paths MDSR maintains. This is suitable for MDMC since the two descriptions are equivalent in importance on video quality. If LC with ARQ is used, an algorithm that always puts the shortest path in cache 0 (which is used by the BL) is preferred.

2) MDSR effectively reduces both the mean packet loss rate and the mean packet loss burst length. When the speed is 10 m/s, the average loss rate for MDSR is about 3%, while that of DSR is about 6.4%. The mean burst length of MDSR at 10 m/s is about 11, while that of DSR is about 20.

3) When the nodes are stationary, the mean burst length of MDSR is 1. Packet loss in this case is mainly caused by failure in accessing the channel. When nodes are mobile, the links are more frequently broken because of nodal mobility, and the loss characteristics change from random loss to bursty loss. We conjecture that the burst length depends on the time scale of the routing protocol and the rate of change in the topology.

4) Somewhat counterintuitive is that as speed increases, the average loss rate becomes stable. Furthermore, the highest mean burst length occurs at 4 m/s instead of 10 m/s. The mean PSNR in Fig. 15 shows the same trend. When speed increases, the mean PSNR first drops, then becomes stable. Note that at 4 m/s, the mean burst length is about 40 packets, which correspond to more than four frames. At this high burst length (and the corresponding high loss rate), the general trend that the PSNR increases with the burst length is not true any more. In fact, the likelihood that both paths experience packet losses simultaneously is quite high, which leads to a significant drop in the decoded video quality.

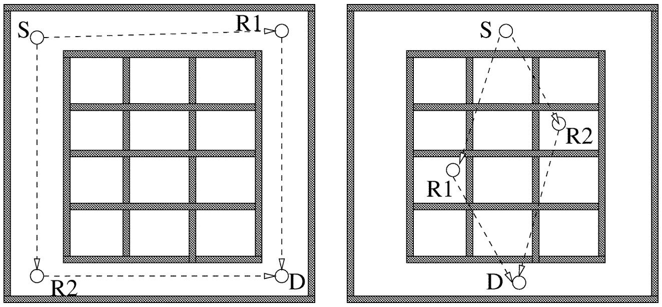
These observations further verify our previous observation that mobility is both harmful and helpful. During the initial increase in mobility, routes break down more easily, which leads to an increase in the mean packet loss rate and a drop in the PSNR. As speed further increases, new topologies are more quickly formed and new routes are more quickly established. A hot spot in the region, where nodes cluster and compete for the channel, is more quickly dispersed. As speed increases, the period of time a node remains disconnected is smaller. The turning point (4 m/s in Figs. 14 and 15) is determined by the node density in the region and the transmission range. We conjecture that similar phenomenon exists for other scenarios with a different number of nodes or a different transmission range, given that the node density is high enough to maintain a connected network for most of the time. When the nodal speed increases even further, the routing process would be unable to track the quickly changing topology. Therefore, drops in the average PSNR is expected.

VI. An AD Hoc Multipath Video Streaming Testbed

To validate the feasibility of video transport over ad hoc networks and evaluate the achievable video quality with today's technology, we implemented an ad hoc multipath video streaming testbed. The implementation and experimental results are reported in the following.

A. Setup of the Testbed

The testbed is implemented using four IBM Thinkpad notebooks with 802.11b cards. Fig. 16 shows the network view of the testbed. The notebooks were placed at (or moved around) various locations in the Library/CATT building at Polytechnic University. IBM High Rate Wireless LAN cards are used working in the DCF mode with a channel bandwidth of 11 Mb/s. The corridors are about 30 m × 60 m. In the building, there is interference from IEEE 802.11 access points (AP) and other electronic devices (e.g., microwave ovens). Nodes S and D are, respectively, the video source and sink, while nodes R_1 and



(a) (b)

Fig. 16. Experiment scenarios for the testbed. (a) Line-of-sight. (b) Behind the walls.

R_2 are the relays. Since there are only four nodes in this network, we use static routing to force the use of two-hop routes. Dynamic routing will be implemented in a future version.

The system is built on Microsoft Windows 2000. We implemented the timestamping, sequence numbering, and QoS feedback functions in the application layer. For LC with ARQ, limited retransmission for BL is implemented in the application layer as well. UDP sockets are used at the transport layer. A traffic allocator dynamically allocates packets to the two paths. The video sink maintains a playout buffer, using a hash table data structure. Typically video streaming applications use a playout buffer of a few seconds to smooth the jitter in incoming packets. We chose a playout buffer of 2 s for this network. To support interactive applications, we also experimented with a 300 ms playout delay.

The implementations of LC with ARQ and MDMC codec discussed in Section IV-A are used. We did not use the RPS codec since it does not support streaming of pre-encoded video. For both schemes used, the testbed performs off-line encoding, but the received video frames are decoded in real time and displayed on the screen of node D.

B. Experimental Results

We examined the performance of the system in the scenarios shown in Fig. 16. The average PSNRs of the received frames using the two schemes are presented in Table II and Table III, respectively. For comparison with the Markov simulation studies presented in Section IV, the last row of each table lists the average PSNRs obtained from Figs. 4–7. Each testbed value in the table is the average over an experiment lasting for 10–15 min. In all the tests using LC with ARQ, path 1 is used for the BL.

The results in Table II are consistent with the Markov simulation results presented in Section IV-C. The minor difference between the last two rows of Table II are caused by the differences in the actual loss rates of the testbed experiments and the corresponding Markov simulations, and the differences in the loss patterns (the experimental loss processes may not be Markovian). An interesting observation is that for the test scenario in Fig. 16(a), when the loss rates for both substreams are very low, MDMC has a higher average PSNR (33.11 dB) than LC with ARQ (32.24 dB). This is also shown in Figs. 5 and 6.

TABLE II
AVERAGE PSNRs OF DECODED FRAMES: MDMC TESTBED EXPERIMENTS/MARKOV SIMULATIONS

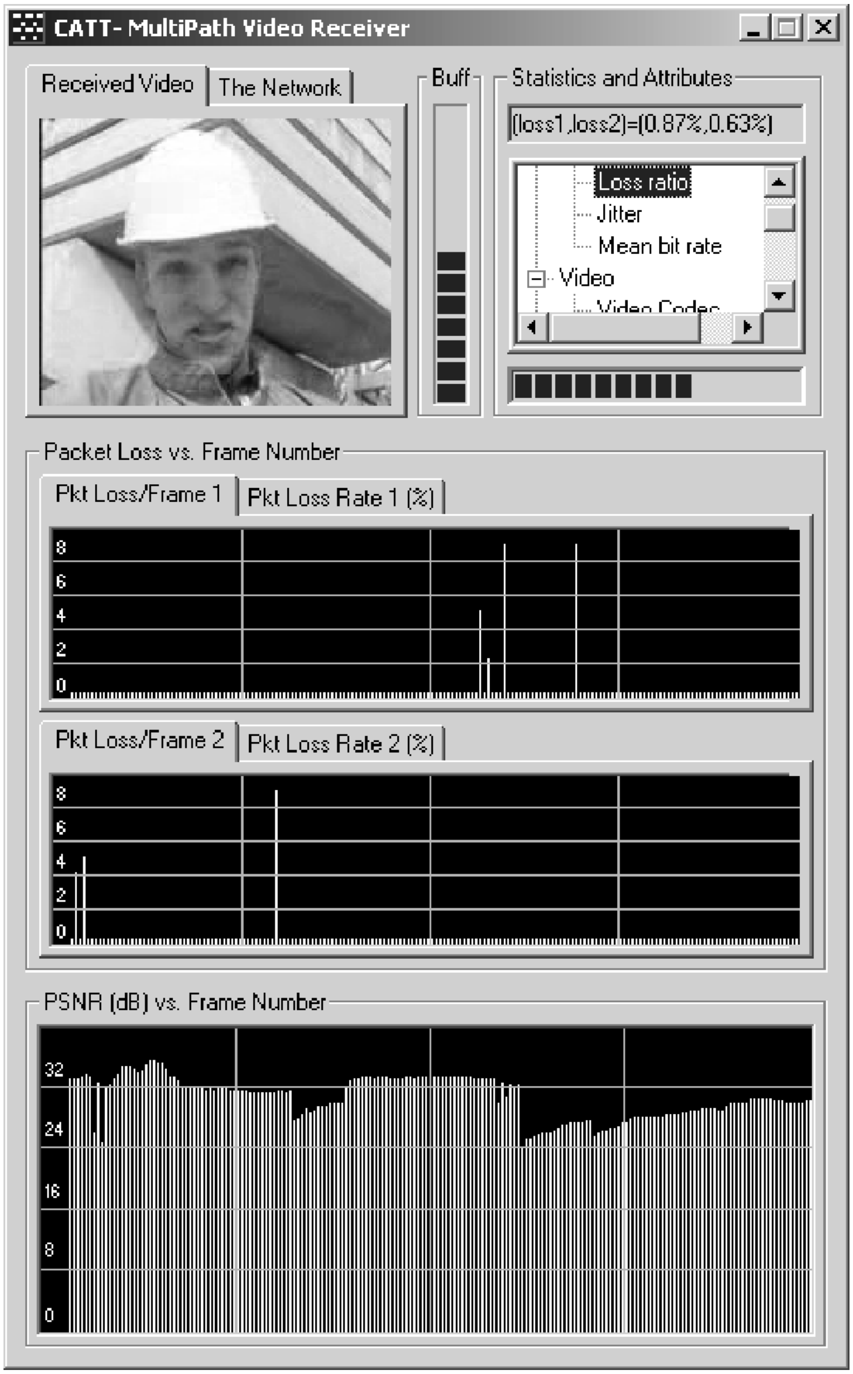
Scenario	Fig.16(a)	Fig.16(b)	Fig.16(b)	Fig.16b)
Playout delay	2s	2s	2s	300ms
Pkt Loss rate 1	0.41%	6.14%	8.46%	8.13%
Pkt Loss rate 2	0.75%	11.96%	7.52%	7.97%
Burst length 1	1.75	3.79	3.67	6.08
Burst length 2	4.76	3.08	3.33	2.34
Ave. PSNR (testbed)	33.11 dB	27.53 dB	28.65 dB	28.16 dB
Ave. PSNR (sim)	33.08 dB	28.05 dB	28.65 dB	28.65 dB

TABLE III
AVERAGE PSNRs OF DECODED FRAMES: LC WITH ARQ TESTBED EXPERIMENTS/MARKOV SIMULATIONS

Scenario	Fig.16(a)	Fig.16(b)	Fig.16(b)	Fig.16(b)
Playout delay	2s	2s	2s	300ms
Ori. BL loss	0.06%	5.95%	7.98%	7.94%
BL loss	0.00%	2.49%	2.25%	5.37%
EL loss	0.38%	12.22%	8.14%	8.16%
Ori. BL Burst len.	3.64	4.80	3.94	4.25
BL Burst len.	0	10.23	6.33	8.58
EL Burst len.	3.29	4.18	3.94	3.16
ARQ succ. ratio	100%	58.0%	71.8%	32.4%
Ave. PSNR (testbed)	32.34 dB	30.64 dB	30.14 dB	30.13 dB
Ave. PSNR (sim)	N/A	31.10 dB	31.68 dB	31.68 dB

The results in Table III clearly show that ARQ effectively reduces the BL packet loss rate in all the experiments. These reductions account for improved video quality. For example, all lost BL packets in the test of Fig. 16(a) were successfully recovered, resulting in a BL loss rate of 0%. The average PSNR for this test is the highest among all the LC with ARQ experiments. However, this is not true for the mean burst length. Since each BL packet has its deadline imposed by the playout delay, ARQ is more effective in recovering short error bursts. During an experiment, many short error bursts are recovered, reducing the number of error bursts. But for an error burst comparable or greater than the playout delay, only a small portion of it is successfully retransmitted. Therefore, sometimes the mean burst length of BL increases when ARQ is used. Recall in Fig. 7, we showed that given the same average loss rate, PSNR improves when the mean burst length increases. The increased mean burst length, combined with reduced average BL loss rate, contributes to the increased video quality. During the experiments with LC with ARQ, we observed short periods of badly corrupted frames, followed by a long period of high quality frames. The ARQ success ratio in the table is defined to be the ratio of the number of successfully retransmitted BL packets and the number of all lost BL packets. This ratio decreases as the loss rates of both paths increase and as the playout delay decreases.

For the experiments reported here, the packet delay and delay jitter are both very low, because there is low background traffic and the bit rates of the video substreams are also low. For this reason, very few packets are dropped because of lateness even with a 300 ms playout delay, and the video equality obtained with a 300 ms playout delay is similar to that with a 2 s playout delay, with both schemes. When the system load



(loss1,loss2)=(0.87%,0.63%)

Fig. 17. A screenshot of the testbed GUI during a MDMC experiment.

is higher, the 300 ms playout delay is likely to yield worse performance.

In the experiments, LC with ARQ performs better than MDMC, except for the very low loss cases, which is consistent with the Markov model simulation results in Figs. 4–7. However, as compared with the results in Figs. 4–7, LC with ARQ yields lower average PSNR under similar loss rates. In fact, a successful retransmission requires: 1) successful and timely delivery of the NACK and 2) successful delivery of the retransmitted BL packet before its deadline. Recall that in the Markov model simulations, we assume a perfect feedback channel and the paths were chosen to be disjoint. These assumptions ensure a high ARQ success ratio. Additionally, in the testbed, NACKs are sent on the same two paths as the video packets. The end-to-end delay is not fixed as in the Markov model simulations. These further reduce the degree of path diversity and the effectiveness of the ARQ algorithm. For MDMC, no feedback is necessary. As a result, its performance is more consistent with the Markov simulation results in Figs. 4–7.

Fig. 17 is a screenshot of the MDMC testbed during an experiment. The upper left part of the GUI displays the received video and the network view of the testbed. The transport related statistics (loss rates, jitter, receiver buffer occupancy, etc.) and the video codec related attributes (frame rate, format, bit rates, etc.)

are displayed at the upper right part of Fig. 17. The two windows in the center display the packet loss traces of the two substreams for each frame. The lower part of Fig. 17 is the PSNR trace of the received video, which illustrates how video quality is impaired by packet losses of both substreams and how the MDMC decoder recovers from the packet losses.

VII. Conclusion

Enabling video transport over ad hoc networks is more challenging than over other wireless networks, both because ad hoc paths are highly unstable and compressed video is susceptible to transmission errors. However, multiple paths in an ad hoc network can be exploited as an effective means to combat transmission errors. Motivated by this observation, we chose three representative video coding schemes, all based on MCP used in modern video coding standards, and show how to adapt these schemes with MPT to enable video transport over ad hoc networks.

In this paper, we highlight the close interaction between the video codec and the transport layer. Results presented suggest that if a feedback channel can be set up, the standard H.263 coder with its RPS option can work quite well. Additionally, if delay caused by one retransmission is acceptable, layered coding with the BL protected by ARQ is more suitable. MDC is the best choice when feedback channels cannot be set up, or when the loss rates on the paths are not too high. A comparison of the schemes from the perspective of video coding, e.g., frame memory required at the coder and decoder, source coding redundancy, and on-line/off-line coding is presented as well. Using OPNET models, we extended DSR to support multipath routing. The impact of multipath routing and mobility is investigated.

To further verify the feasibility of video transport over ad hoc networks, we implemented an ad hoc multipath video streaming testbed using four notebook computers. Our tests show that acceptable quality streaming video is achievable with both LC with ARQ and MDMC, in the range of video bit rate, background traffic, and motion speed examined. Together with simulation results using the Markov path model and the OPNET models, our studies demonstrate the viability of video transport over ad hoc networks using multipath transport and multistream coding.

We should note that further improvements could be made for each component in the proposed framework. For example: 1) the video codec parameters could be further tuned and optimized in the rate distortion sense, given the path conditions; 2) packets from all the substreams could be dispersed to the multiple paths with a more sophisticated algorithm to maximize the benefit of MPT; and 3) the ad hoc networks simulated in Markov model and OPNET models, and the testbed are relative small. It would be very interesting to see how the performance scales for a larger ad hoc network. These are still open research problems that are worth investigating in future work.

Acknowledgment

The authors would like to thank the anonymous reviewers for their helpful comments which improved the quality of this paper. We also thank the Wireless Communications Technologies Group of the National Institute of Standards and Technology for making the OPNET DSR model available in the public domain.

References

[1] *Draft Standard for Wireless LAN: Medium Access Control (MAC) and Physical Layer (PHY) Specifications IEEE*, IEEE Standard 802.11, July 1996.

[2] C. R. Lin and M. Gerla, “Asynchronous multimedia multihop wireless networks,” in *Proc. IEEE/ACM INFOCOM*, Kobe, Japan, Apr. 1997, pp. 118–125.

[3] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, “Error resilient video coding techniques,” *IEEE Signal Processing Mag.*, vol. 17, pp. 61–82, July 2000.

[4] Y. Wang and Q.-F. Zhu, “Error control and concealment for video communication: A review,” *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.

[5] N. Gogate, D. Chung, S. S. Panwar, and Y. Wang, “Supporting image/video applications in a multihop radio environment using route diversity and multiple description coding,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 12, no. 9, pp. 777–792, Sept. 2002.

[6] S. Lin, S. Mao, Y. Wang, and S. S. Panwar, “A reference picture selection scheme for video transmission over ad hoc networks using multiple paths,” in *Proc. IEEE ICME*, Tokyo, Japan, Aug. 2001, pp. 96–99.

[7] S. Mao, S. Lin, S. S. Panwar, and Y. Wang, “Reliable transmission of video over ad hoc networks using automatic repeat request and multipath transport,” in *Proc. IEEE Fall VTC*, Atlantic City, NJ, Oct. 2001, pp. 615–619.

[8] Y. Wang and S. Lin, “Error resilient video coding using multiple description motion compensation,” *IEEE Trans. Circuits Syst. Video Technol*, vol. 12, no. 6, pp. 438–452, 2002.

[9] E. N. Gilbert, “Capacity of a bursty-noise channel,” *Bell Syst. Tech. J.*, vol. 39, no. 9, pp. 1253–1265, Sept. 1960.

[10] OPNET Modeler. OPNET Tech., Inc. [Online]. Available: http://www.mil3.com

[11] S. Wenger, “Video redundancy coding in H.263+,” in Workshop Audio-Visual Services for Packet Networks, Aberdeen, Scotland, Sept. 1997.

[12] M. Khansari, A. Jalali, E. Dubois, and P. Mermelstein, “Low bit-rate video transmission over fading channels for wireless microcellular systems,” *IEEE Trans. Circuit Syst. Video Technol.*, vol. 6, pp. 1–11, Feb. 1996.

[13] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for realtime applications. presented at IETF request for comments 1889. [Online]. Available: http://www.ietf.org

[14] V. K. Goyal, “Multiple description coding: Compression meets the network,” *IEEE Signal Processing Mag.*, vol. 18, pp. 74–93, Sept. 2001.

[15] A. Reibman, H. Jafarkhani, Y. Wang, M. Orchard, and R. Puri, “Multiple description coding for video using motin compensated prediction,” in *Proc. IEEE ICIP*, Kobe, Japan, Oct. 1999, pp. 837–841.

[16] J. G. Apostolopoulos, “Error-resilient video compression through the use of multiple states,” in *Proc. IEEE ICIP*, Vancouver, BC, Canada, Sept. 2000, pp. 352–355.

[17] M. de Cordeiro *et al.*, “Establishing a trade-off between unicast and multicast retransmission modes for reliable multicast protocols,” in *Proc. Int. Symp. Modeling, Analysis, Simulation of Computer Telecommunication Systems*, 2000, pp. 85–91.

[18] N. F. Maxemchuck, “Diversity routing,” in *Proc. IEEE ICC*, vol. 1, San Francisco, CA, June 1975, pp. 10–41.

[19] T. Wu and R. Lau, “A class of self-healing ring architectures for SONET network applications,” *IEEE Trans. Commun.*, vol. 40, pp. 1746–1756, Nov. 1992.

[20] R. R. Stewart and Q. Xie, *Stream Control Transmission Protocol: A Reference Guide*. Reading, MA: Addison-Wesley, 2001.

[21] D. Sidhu, R. Nair, and S. Abdallah, “Finding disjoint paths in networks,” in *Proc. ACM SIGCOMM*, Zurich, Switzerland, Sept. 1991, pp. 43–51.

[22] S.-J. Lee and M. Gerla, "Split multipath routing with maximally disjoint paths in ad hoc networks," in *Proc. IEEE ICC*, Helsinki, Finland, June 2001, pp. 3201–3205.

[23] P. Papadimitratos, Z. J. Haas, and E. G. Sirer, "Path set selection in mobile ad hoc networks," in *Proc. ACM MOBIHOC*, Lausanne, Switzerland, June 2002, pp. 1–11.

[24] A. Nasipuri, R. Castañeda, and S. R. Das, "Performance of multipath routing for on-demond protocols in mobile ad hoc networks," *Mobile Networks and Applications*, vol. 6, no. 4, pp. 339–349, 2001.

[25] S. Nelakuditi, Z.-L. Zhang, and R. P. Tsang, "Adaptive proportional routing: A localized QoS routing approach," in *Proc. IEEE INFOCOM*, Tel-Aviv, Israel, Mar. 2000, pp. 1566–1575.

[26] J. Moy. OSPF version 2. presented at IETF Request for Comments 2328. [Online]. Available: http://www.ietf.org

[27] E. Gustafsson and G. Karlsson, "A literature survey on traffic dispersion," *IEEE Network*, pp. 28–36, Mar./Apr. 1997.

[28] D. B. Johnson, D. A. Maltz, Y.-C. Hu, and J. G. Jetcheva, "The dynamic source routing protocol for mobile Ad Hoc networks," in IETF Internet Draft (draft-ietf-manet-dsr-03.txt), Oct. 1999.

[29] C. E. Perkins, E. M. Royer, and S. R. Dos, Ad hoc on-Demand Distance Vector (AODV) Routing, IETF Internet Draft (draft-ietf-manet-aodv-06.txt), July 2000.

[30] Z. J. Haas and M. R. Pearlman, "The zone routing protocol (ZRP) for ad hoc networks," in IETF Internet Draft (draft-ietf-manet-zone-zrp02.txt), June 1999.

[31] M. Pearlman, Z. J. Hass, P. Sholander, and S. S. Tabrizi, "On the impact of alternate path routing for load balancing in mobile ad hoc networks," in *Proc. MobiHOC*, Aug. 2000, pp. 3–10.

[32] N. Gogate and S. S. Panwar, "On a resequencing model for high speed networks," in *Proc. IEEE INFOCOM*, June 1994, pp. 40–47.

[33] Y. Nebat and M. Sidi, "Resequencing considerations in parallel downloads," in *Proc. IEEE INFOCOM*, June 2002, pp. 1326–1336.

[34] N. Gogate and S. S. Panwar, "Supporting applications in a mobile multihop radio enviroment using route diversity: I. Nonrealtime data," in *Proc. IEEE ICC*, June 1998, pp. 802–806.

[35] D. S. Phatak and T. Goff, "A novel mechanism for data streaming across multiple IP links for improving throughput and reliabilit in mobile environments," in *Proc. IEEE INFOCOM*, June 2002, pp. 773–782.

[36] H.-Y. Hsieh and R. Sivakumar, "A transport layer approach for achieving aggregate bandwidths on multi-homed mobile hosts," in *Proc. ACM Int. Conf. Mobile Computer Networking*, Sept. 2002, pp. 83–95.

[37] N. Gogate and S. S. Panwar, "Supporting video/image applications in a mobile multihop radio environment using route diversity," in *Proc. IEEE ICC*, June 1999, pp. 1701–1706.

[38] J. G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," in *Proc. SPIE Conf. Visual Communication Image Processing*, Jan. 2001, pp. 392–409.

[39] J. G. Apostolopoulos, T. Wong, W. Tan, and S. Wee, "On multiple description streaming in content delivery networks," in *Proc. IEEE INFOCOM*, June 2002, pp. 1736–1745.

[40] T. Nguyen and A. Zakhor, "Path diversity with forward error correction (PDF) system for packet switched networks," in IEEE INFOCOM, San Francisco, CA, Apr. 2003.

[41] Y. J. Liang, E. G. Steinbach, and B. Girod, "Multi-stream voice over IP using packet path diversity," in *Proc. IEEE Multimedia Signal Processing Workshop*, Sept. 2001, pp. 555–560.

[42] Y. J. Liang, E. Setton, and B. Girod, "Channel-adaptive video streaming using packet path diversity and rate-distortion optimized reference picture selection," in The IEEE Multimedia Signal Processing Workshop, St. Thomas, U.S. Virgin Islands, Dec. 2002, pp. 420–423.

[43] V. Jacobson, "Congestion avoidance and control," *Comput. Commun. Rev.*, vol. 19, no. 4, pp. 314–329, Aug. 1988.

[44] D. A. Maltz, J. Broch, and D. B. Johnson, "Lessons from a full-scale multi-hop wireless ad hoc network testbed," *IEEE Pers. Commun.*, vol. 8, pp. 8–15, Feb. 2001.

[45] H. Lundgren *et al.*, "A large-scale testbed for reproducible ad hoc protocol evaluations," in *Proc. IEEE WCNC*, 2002, pp. 412–418.

[46] W. Kellerer, E. Steinbach, P. Eisert, and B. Girod, "A real-time internet streaming media testbed," in *Proc. IEEE ICME*, Aug. 2002, pp. 453–456.

[47] W. R. Stevens, *TCP/IP Illustrated, Volume I: The Protocols. Reading*, MA: Addison-Wesley, 1994.

[48] "Video Coding for Low Bit Rate Communication," ITU-T Recommendation H.263, 1998.

[49] $\mathrm{H.263+}$ Codec. University of British Columbia. [Online]. Available: ftp://dspftp.ece.ubc.ca

[50] A. Nasipuri and S. R. Das, "On-demand multipath routing for mobile Ad hoc networks," in *Proc. ICCCN*, 1999, pp. 64–70.

[51] OPNET DSR Model. The Wireless Communications Technologies Group of the National Institute of Standards and Technology. [Online]. Available: http://w3.antd.nist.gov/wctg/prd_dsrfiles.html

[52] T. K. Philips, S. S. Panwar, and A. N. Tantawi, "Connectivity properties of a packet radio network model," *IEEE Trans. Inform. Theory*, vol. 35, pp. 1044–1047, Sept. 1989.

[53] J. Broch, D. A. Maltz, D. B. Johnson, Y.-C. Hu, and J. Jetcheva, "A performance comparison of multi-hop wireless ad hoc network routing protocols," in *Proc. ACM/IEEE Int. Conf. Mobile Computers and Networking*, 1998, pp. 85–97.

[54] J. Yoon, M. Liu, and B. Noble, "Random waypoint considered harmful," in Proc. INFOCOM, San Francisco, CA, Apr. 2003, pp. 1312–1321.

Shiwen Mao (S'99) received the B.S. degree in electrical engineering, the B.E. degree in enterprise management, the M.S. degree in electrical engineering from Tsinghua University, Beijing, China, in 1994 and 1997, respectively, and the M.S. degree in system engineering from Polytechnic University, Brooklyn, NY, in 2000, where he is currently working toward the Ph.D. degree.

From 1994 to 1995, he was a Research Assistant at the State Key Lab on Microwave and Digital Communications, Beijing, China. From 1997 to 1998, he was a Research Member of IBM China Research Laboratory. In the summer of 2001, he was a Research Intern in Avaya Labs-Research, Holmdel, NJ. His research interests include real-time transport in the Internet and wireless networks, performance of wireless ad hoc networks, and queueing analysis.

Shunan Lin (S'00) received the B.S. degree from University of Science and Technology of China, Hefei, and the M.S. degree from the Institute of Automation, Chinese Academy of Sciences, both in electrical engineering. He is working toward the Ph.D. degree at Polytechnic University, Brooklyn, NY.

From 2000 and 2002, he was an Internship Student in Microsoft Research, Beijing, China, and Mitsubishi Electric Research Lab, Murray Hill, NJ, respectively. His research interests are in the field of video singal processing, transmission, and motion estimation.

Shivendra S. Panwar (S'82–M'85–SM'00) received the B.Tech. degree in electrical engineering from the Indian Institute of Technology, Kanpur, in 1981, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Massachusetts, Amherst, in 1983 and 1986, respectively.

He is a Professor in the Electrical and Computer Engineering Department, Polytechnic University, Brooklyn, NY. He is currently the Director of the New York State Center for Advanced Technology in Telecommunications (CATT). He spent the summer of 1987 as a Visiting Scientist at the IBM T. J. Watson Research Center, Yorktown Heights, NY, and has been a Consultant to AT&T Bell Laboratories, Holmdel, NJ. His research interests include the performance analysis and design of networks. Current work includes protocol analysis, traffic and call admission control, switch performance, and multimedia transport over wireless networks.

Dr. Panwar has served as the Secretary of the Technical Affairs Council of the IEEE Communications Society (1992–1993) and is a Member of the Technical Committee on Computer Communications. He is a coeditor of *Network Management and Control, Vol. II* and *Multimedia Communications and Video Coding* (New York: Plenum, 1994 and 1996).

Yao Wang (M'90–SM'98) received the B.S. and M.S. degrees in electronic engineering from Tsinghua University, Beijing, China, in 1983 and 1985, respectively, and the Ph.D. degree in electrical and computer engineering from University of California at Santa Barbara, in 1990.

Since 1990, she has been with the faculty of Polytechnic University, Brooklyn, NY, and is presently Professor of electrical and computer engineering. She was on sabbatical leave at Princeton University, Princeton, NJ, in 1998, and was a Visiting Professor at University of Erlangen, Erlangen, Germany, in the summer of 1998. She was a Consultant with AT&T Labs—Research, Middletown, NJ, formerly AT&T Bell Laboratories, from 1992 to 2000. Her research areas include video communications, multimedia signal processing, and medical imaging. She is the leading author of the textbook *Video Processing and Communications* and has published over 100 papers in journals and conference proceedings.

Dr. Wang has served as an Associate Editor for IEEE TRANSACTIONS ON MULTIMEDIA and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. She received New York City Mayor's Award for Excellence in Science and Technology in the Young Investigator Category in 2000.

Emre Celebi (S'96) received the B.Sc. degree in computer engineering and the M.Sc. degree in system and control engineering from Bogazici University, Istanbul, Turkey, in 1998 and 2001, respectively. He is currently a Research Fellow and is working toward the Ph.D. degree in computer science at Polytechnic University, Brooklyn, NY.

His research interests include wireless networks, performance analysis, evolutionary and adaptive protocols.