Distortion-Resilient Routing for Video Flows in Wireless Multi-hop Networks

George Papageorgiou*, Shailendra Singh*, Srikanth V. Krishnamurthy*, Ramesh Govindan† and Tom La Porta‡

*UC Riverside, {*gpapag,singhs,krish*}*@cs.ucr.edu*, †USC, *ramesh@usc.edu*, ‡Penn State University, *tlp@cse.psu.edu*

Abstract—**Traditional routing metrics designed for wireless networks are application agnostic. In this paper, we consider a wireless network where the application flows consist of video traffic. From a user-perspective, reducing the level of video distortion is critical. We ask the question "Should the routing policies change if the end-to-end video distortion is to be minimized?" Popular link-quality based routing metrics (such as ETX) do not account for dependence (in terms of congestion) across the links of a path; as a result, they can cause video flows to converge onto a few paths and thus, cause high video distortion. To account for the evolution of the video frame loss process we construct an analytical framework to first, understand and second, assess the impact of the wireless network on video distortion. The framework allows us to formulate a routing policy for minimizing distortion, based on which we design a protocol for routing video traffic. We find via simulations and testbed experiments that our protocol is efficient in reducing video distortion and minimizing the user experience degradation. Specifically, our protocol reduces the distortion by 20% over traditional methods, which significantly improves the video quality perceived by a user.**

I. INTRODUCTION

With the advent of smartphones video traffic has become very popular in wireless networks. In tactical networks or disaster recovery, one can envision the transfer of video clips to facilitate mission management. From a user perspective, maintaining a good quality of the transferred video is critical. The video quality is affected by (i) the distortion due to compression at the source and (ii) the distortion due to both wireless channel induced errors and interference.

Video encoding standards, like MPEG-4 [1] or H.264/AVC [2] define groups of *I*, *P* and *B* type frames that provide different levels of encoding and thus, protection against transmission losses. This Group of Pictures (GOP) allows for the mapping of frame losses into a distortion metric which can be use to assess the application level performance of video transmissions.

One of the critical functionalities which is often neglected, but affects the end-to-end quality of a video flow is routing. Typical routing protocols, designed for wireless multi-hop settings are application agnostic and do not account for correlation of losses on the links that compose a route from a source to a destination node. Furthermore, since flows are considered independently they can converge on to certain links which then become heavily loaded (thereby increasing video distortion), while others are significantly underutilized. The decisions made by such routing protocols are based on only network (and not application) parameters.

In this paper, our thesis is that the user perceived video quality can be significantly improved by accounting for application requirements, and specifically the video distortion experienced by a flow end-to-end. Typically, the schemes used to encode a video clip can accommodate a certain number of packet losses per frame. However, if the number of lost packets in a frame exceeds a certain threshold, the frame cannot be decoded correctly. A frame loss will result in some amount of distortion. The value of distortion at a hop along the path from the source to the destination depends on the positions of the unrecoverable video frames (simply referred to as frames) in the GOP, at that hop. As one of our main contributions, we construct an analytical model to characterize the dynamic behavior of the process that describes the evolution of frame losses in the GOP (instead of just focusing on a network quality metric such as the packet loss probability) as video is delivered on an end-to-end path. Specifically, with our model we capture how the choice of path for an end-to-end flow affects the performance of a flow in terms of video distortion. The packet loss probability on a link is mapped to the probability of a frame loss in the GOP. The frame loss probability is then directly associated with the video distortion metric. By using the above mapping from the network specific property (i.e. packet loss probability) to the application specific quality metric (i.e. video distortion) we pose the problem of routing as an optimization problem where the objective is to find the path from the source to the destination that minimizes the end-to-end distortion.

In our formulation, we explicitly take into account the history of losses in the GOP along the path. This is in stark contrast with traditional routing metrics (such as the total Expected Transmission Count (ETX) [3]) wherein, the links are treated independently. Our solution to the problem is based on a dynamic programming approach which effectively captures the evolution of the frame loss process. We then design a practical routing protocol, based on the above solution, to minimize routing distortion. In a nutshell, since the loss of the longer *I* frames that carry fine-grained information affect the distortion metric more, our approach ensures that these frames are carried on the paths that experience the least congestion; the latter frames in a GOP are sent out on relatively more congested paths.

Specifically, our contributions in this paper are as follows:

- **Developing an analytical framework to capture the impact of routing on video distortion:** As our primary contribution, we develop an analytical framework that captures the impact of routing on the end-to-end video quality in terms of distortion. Specifically, the framework facilitates the computation of routes that are optimal in terms of achieving the minimum distortion. The model takes into account the joint impact of the PHY and MAC layers and the application semantics on the video quality.
- **Design of a practical routing protocol for distortion-resilient video delivery:** Based on our analysis, we

design a practical routing protocol for a network that primarily carries wireless video. The practical protocol allows a source to collect distortion information on the links in the network, and distributing traffic across the different paths in accordance to (a) the distortion and (b) the position of a frame in the GOP.

- **Evaluations via extensive experiments:** We demonstrate via extensive simulations and real testbed experiments on a multi-hop 802.11a testbed that our protocol is extremely effective in reducing the end-to-end video distortion and keeping the user experience degradation to a minimum. In particular, the use of the protocol increases the PSNR of video flows by as much as 20% producing flows with an MOS that is on the average 2-3 times higher compared to the case when traditional routing schemes are used. These PSNR and MOS gains project significant improvements in the perceived video quality at the destination of a flow [4]. We also evaluate our protocol with respect to various system parameters.

Organization: The paper is organized as follows. Related work is presented in Section II. Our analytical models are in Section III, followed by the problem formulation in Section IV. In Section V, we discuss how our framework can be used to route video in practice. Section VI contains results from our simulations and testbed experiments. We conclude in Section VII.

II. RELATED WORK

The plethora of recommendations from the standardization bodies regarding the encoding and transmission of video indicates the significance of video communications. Standards like the MPEG-4 [1] and the H.264/AVC [2] provide guidelines on how a video clip should be encoded for a transmission over a communication system. Typically, the initial video clip is separated into a sequence of frames of different importance with respect to quality and hence, different levels of encoding. The frames are called *I*, *P* and *B* frames and groups of such frames constitute a structure named the Group of Picture (GOP). In each such GOP, the first frame is an I-frame which can be decoded independently of any other information carried within the same GOP. After the I-frame a sequence of P and possibly B-frames follows. The P and B-frames use the I-frame as a reference to encode information.

There has been a body of work on packet loss resilient video coding in the singal processing research community [5]. In [4], the video stream is split into high and low priority partitions and FEC is used to protect the high priority data. To account for temporal and spatial error propagation due to quantization and packet losses, an algorithm is proposed in [6] to produce estimates of the overall video distortion that can be used for switching between inter and intra coding modes per macroblock, achieving higher PSNR. In [7], an enhancement to the transmission robustness of the coded bitstream is achieved through the introduction of inter/intra-coding with redundant macroblocks. The coding parameters are determined by a rate-distortion optimization scheme. These schemes are evaluated using simulation where the effect of the network transmission is represented by a constant packet loss rate, and therefore fails to capture the idiosyncrasies of real world systems.

In [8], an analytical framework is developed to model the effects of wireless channel fading on video distortion. The model is however, only valid for single hop communication. In [9], the authors examine the effects of packet loss patterns and specifically the length of error bursts, on the distortion of compressed video. The work, although on a single link, showcases the importance of accounting for the correlation of errors across frames. The performance of video streaming over a multi-hop IEEE 802.11 wireless network is studied in [10], and a two-dimensional Markov chain model is proposed. The model is used not only for performance evaluation but also as a guide for deploying video streaming services with end-to-end QoS provisioning. Finally, a recursion model is derived in [11] to relate the average transmission distortion across successive P-frames. None of these efforts consider the impact of routing on video distortion.

There have also been studies on the performance of video transmissions over 4G wireless networks that have been designed to support high Quality of Service (QoS) for multimedia applications. In [12] an assessment of the recently defined video coding scheme (H.264/SVC) is performed over mobile WiMAX. Metrics such as the Peak Signal to Noise Ratio (PSNR) and the Mean Opinion Score (MOS) are used to represent the quality of experience perceived by the end user. The results show that the performance is sensitive to the different encoding options in the protocols and respond differently to the loss of data in the network. Again, these are single link wireless networks and routing is not a factor.

In [13], a multi-path routing scheme for video delivery over IEEE 802.11 based wireless mesh networks is proposed. To achieve good traffic engineering the scheme relies on maximally disjoint paths. However, this work does not consider distortion as a user-perceived metric. It simply aims to reduce the latency of video transmissions, and thus, its objective is different from what we consider here.

III. MODEL FORMULATION

Our analytical model couples the functionality of the physical and MAC layers of the network with the application layer for a video clip that is sent from a source to a destination node. The model for the lower layers computes the packet loss probability through a set of equations that characterize multiuser interference, physical path conditions and traffic rates between source-destination pairs in the network. This packet loss probability is then input to a second model to compute the frame loss probability and from that the corresponding distortion. The value of the distortion at a hop along the path from the source to the destination node depends on the position of the first unrecoverable frame in the GOP.

A. The PHY and MAC Layer Modelling

We consider an IEEE 802.11 network of N nodes denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. For each node $i \in \mathcal{N}$, denote by $\mathcal{P}_i$ the set of paths that pass via node i. For simplicity, we assume a constant packet length of L bits for all source-destination paths. There are various models [14]–[17] that attempt to capture the operations of the IEEE 802.11 protocol. We use the model in [17] to represent the operations of the PHY and MAC layers. Due to space constraints we only briefly outline the model, specifics can be found in [17].

The approach followed in [17] is based on network loss models. Three sets of equations are derived. The first corresponds to a scheduling model that computes the serving rate $\rho_{i,p}$ of a path p at each node i, as a function of the scheduler coefficient $k_{i,p}$ and the service time $T_{i,p}$:

$$\rho_{i,p} = k_{i,p}\mathrm{E}[T_{i,p}] \tag{1}$$

The second captures the IEEE 802.11 MAC and PHY models and associates the probability $\beta_{i,p}$ of a transmission failure with the channel access probability $\alpha_{i,p}$

$$\alpha_{i,p} = \frac{2(1 - 2\beta_{i,p})}{W(1 - 2\beta_{i,p}) + \beta_{i,p}(W + 1)(1 - (2\beta_{i,p})^B)}, \tag{2}$$

where B is the number of backoff stages and W is the minimum window size. Finally, the third set of equations describes the routing model and computes the incoming traffic rate $\lambda_{j,p}$ to the next hop node j based on scheduling and transmission failures:

$$\lambda_{j,p} = k_{i,p}(1 - \beta_{i,p}), \quad \text{for all } i, j \in \mathcal{N}, p \in \mathcal{P}_i. \tag{3}$$

A fixed point method is used to couple the equations in an iteration, until convergence to a consistent solution is achieved and satisfied. The solution is an approximation to the packet loss probability β per link and the throughput of the network. Note here that any other method can be used to find β, which can then be used in our video distortion estimation framework described below.

B. Video Distortion Model

Our analysis is based on the model for video transmission distortion in [8]. The distortion is broken down into source distortion and wireless transmission distortion over a single hop. Instead of focusing on a single hop, we significantly extend the analysis by developing a model that captures the evolution of the transmission distortion along the links of a route from the source node to the destination node.

We consider a Group of Pictures (GOP) structure which consists of an I-frame followed by $(F-1)$ P-frames. We index each frame in the GOP structure starting from 0, i.e., the I-frame corresponds to index 0, and the P-frames correspond to indices from 1 up to $(F-1)$. We focus on predictive source coding where, if the ith frame is the first lost frame in a GOP, then the ith frame and all its successors in the GOP are replaced by the $(i-1)^{st}$ frame at the destination node. Assuming that the sequence of frames is stationary, the average distortion introduced by such a frame replacement depends on the temporal proximity of the replaced frame to the $(i-1)^{st}$ frame and not on the actual position of the frame (in the GOP) to be replaced. In [8], a linear model, that corresponds to empirical data, is used to provide the average Mean Squared Error (MSE) as a function of the temporal distance between frames. Using this model, the average distortion $D^{(i)}$ is computed in [8] to be:

$$D^{(i)} = (F - i) \cdot \frac{i \cdot F \cdot D^{\min} + (F - i - 1) \cdot D^{\max}}{(F - 1) \cdot F} \tag{4}$$

for $i = 0, 1, 2, \ldots, (F-1)$. The minimum distortion $D^{\min} = D^{(F-1)}$ is achieved when the last frame in the GOP is lost and the maximum, $D^{\max} = D^{(0)}$ is attained if the first frame

is lost. The values $D^{\min}$ and $D^{\max}$ depend on the actual video sequence and have to be determined by measurement.

If R_s is the source coding rate, α_I and α_P are the percentage of bits in the GOP that belong to an I-frame and a P-frame, respectively, then

- the number of packets per an I-frame $= \alpha_I R_s \frac{T_{\mathrm{GOP}}}{L}$, and
- the number of packets per a P-frame $= \alpha_P R_s \frac{T_{\mathrm{GOP}}}{L}$,

where $\alpha_I + (F-1)\alpha_P = 1$ and T_{GOP} is the duration of a GOP. We define the *sensitivity* of a frame to lost packets to be the minimum number of packets that belong to a frame which if lost, can prevent the correct decoding of the frame. We denote by s_I the sensitivity of an I-frame and by s_P, that of a P-frame. For the sensitivity of the I-frame it holds that $1 \le s_I \le \alpha_I R_s \frac{T_{\mathrm{GOP}}}{L}$ and for the P-frame it is $1 \le s_P \le \alpha_P R_s \frac{T_{\mathrm{GOP}}}{L}$. Note, that any further packet losses beyond s_I for the I-frame and s_P for the P-frame do not cause any additional distortion for that particular GOP, because in that case the corresponding frame is already considered lost and cannot be correctly decoded.

We extend the wireless transmission distortion introduced in [8] and defined in (4) for the multi-hop case. We define the sequence $D = \{D_t, \ t = 0, 1, \ldots\}$ to represent the wireless transmission distortion along the path from the source to the destination, where D_t is the wireless transmission video distortion at the tth-hop. In general, at the tth-hop the distortion D_t can take one of the following discrete values given by (4):

$$\left\{D^{(0)}, D^{(1)}, \ldots, D^{(F-1)}\right\} \cup \{0\}. \tag{5}$$

The sequence of values the process D takes depends on the number of lost packets per frame in the GOP at each link. Clearly, $D_t \le D_{t+1}$ w.p.1, for all t.

We track the packet losses per frame by defining a multi-dimensional counting process $\boldsymbol{M} = \{\boldsymbol{M}_t, \ t = 0, 1, \ldots\}$:

$$\boldsymbol{M}_t = \left(M_t^{(0)}, M_t^{(1)}, \ldots, M_t^{(F-1)}\right) \tag{6}$$

where, the index t is again the hop count along the path from the source to the destination. The first component $M_t^{(0)}$ of the counting process $\boldsymbol{M}$ tracks the number of lost packets from the I-frame at the tth hop along the path and the components $M_t^{(1)}, \ldots, M_t^{(F-1)}$ count the lost packets that belong to the subsequent P-frames in the GOP at the tth hop. The state space for each of these components is given by (7) and (8):

$$M_t^{(0)} \in \{0, 1, \ldots, s_I\}, \quad t = 0, 1, \ldots \tag{7}$$

$$M_t^{(k)} \in \{0, 1, \ldots, s_P\}, \quad t = 0, 1, \ldots \tag{8}$$

for $k = 1, 2, \ldots, (F-1)$.

Assuming that the packet losses in different frames in the GOP are independent events (likely if the fading patterns change in between), the transition probabilities for the process $\boldsymbol{M}$, can be computed. Suppose that β is the packet loss probability provided by the analytical model that describes the MAC layer (in Section III-A). Furthermore, let the value of $\boldsymbol{M}$ at hop t be $\boldsymbol{M}_t = (i_0, i_1, i_2, \ldots, i_{(F-1)})$ and at hop $(t + 1)$ be $\boldsymbol{M}_{t+1} = (j_0, j_1, j_2, \ldots, j_{(F-1)})$. Since each of the components of $\boldsymbol{M}$ is a counting process, the corresponding sample paths are non-decreasing w.p.1, and therefore

$i_k \le j_k$, $k = 0, 1, \ldots, (F-1)$. Regarding the transitions of $M^{(0)}$ that corresponds to the I-frame, we have that

$$0 \le i_0 \le j_0 \le s_I \implies j_0 - i_0 \in \{0, 1, \ldots, s_I - i_0\} \quad (9)$$

The corresponding transition probability $\phi_{i_0 j_0}^{(0)}$ is equal to the probability of losing $(j_0 - i_0)$ packets out of the $(s_I - i_0)$ remaining packets in the I-frame. Therefore, the transition probabilities for the first component $M^{(0)}$ are given by the following binomial distribution:

$$\phi_{i_0 j_0}^{(0)} = P\left\{ M_{t+1}^{(0)} = j_0 \mid M_t^{(0)} = i_0 \right\}$$
$$= \begin{cases} \binom{s_I - i_0}{j_0 - i_0} \beta^{j_0 - i_0} (1 - \beta)^{s_I - j_0}, & j_0 = i_0, \ldots, \\ & \qquad \ldots, s_I \quad (10) \\ 0, & \text{otherwise.} \end{cases}$$

Similar to the transitions of $M^{(0)}$, the transitions of $M^{(k)}$, $k = 1, 2 \ldots, (F-1)$ that correspond to the P-frames in the GOP are specified by the transition probabilities:

$$\phi_{i_k j_k}^{(k)} = P\left\{ M_{t+1}^{(k)} = j_k \mid M_t^{(k)} = i_k \right\}$$
$$= \begin{cases} \binom{s_P - i_k}{j_k - i_k} \beta^{j_k - i_k} (1 - \beta)^{s_P - j_k}, & j_k = i_k, \ldots, \\ & \qquad \ldots, s_P \quad (11) \\ 0, & \text{otherwise.} \end{cases}$$

From the transition probabilities (10) and (11) one can compute the distribution $\pi_t^{(k)}, k = 0, 1, \ldots, (F-1)$ of lost packets in each frame at hop t assuming that there are no lost packets at the source. In particular, for the I-frame we have:

$$\pi_t^{(0)}(j) = P\left\{ M_t^{(0)} = j \mid M_0^{(0)} = 0 \right\} = \sum_{i=0}^{s_I} \phi_{ij}^{(0)} \pi_{t-1}^{(0)}(i), \quad (12)$$

for $j = 0, 1, \ldots, s_I$. We define the row vector

$$\boldsymbol{\pi}_t^{(0)} = \left(\pi_t^{(0)}(0), \pi_t^{(0)}(1), \ldots, \pi_t^{(0)}(s_I) \right), \quad (13)$$

Then (12), in vector form, becomes:

$$\boldsymbol{\pi}_t^{(0)} = \boldsymbol{\pi}_{t-1}^{(0)} \Phi_I, \quad (14)$$

for $t = 1, 2, \ldots$, where $\Phi_I = [\phi_{ij}^{(0)}]$ is the transition matrix for the process $M^{(0)}$. To make the dependence of the matrix Φ_I to the packet loss probability β explicit, we use the notation $\Phi_I(\beta)$.

It follows then from (14) that:

$$\boldsymbol{\pi}_t^{(0)} = \boldsymbol{\pi}_0^{(0)} \cdot \Phi_I(\beta_1) \cdots \Phi_I(\beta_t), \quad t = 1, 2, \ldots, \quad (15)$$

for a sequence of packet loss probabilities β_i, $i = 1, 2, \ldots$, where $\boldsymbol{\pi}_0^{(0)} = (1, 0, \ldots, 0)$. Following the same process, we can compute the corresponding distribution $\boldsymbol{\pi}_t^{(k)}$ for the kth P-frame in the GOP:

$$\boldsymbol{\pi}_t^{(k)} = \boldsymbol{\pi}_0^{(k)} \cdot \Phi_P(\beta_1) \cdots \Phi_P(\beta_t), \quad t = 1, 2, \ldots, \quad (16)$$

where $\boldsymbol{\pi}_0^{(k)} = (1, 0, \ldots, 0)$ and $\Phi_P = [\phi_{ij}^{(k)}]$ is the transition matrix for the process $M^{(k)}$, $k = 1, 2, \ldots, (F-1)$. As one can immediately see, the packet loss probabilities, computed after accounting for the PHY and MAC, in the previous section, can be used here to compute the probabilities $\boldsymbol{\pi}_t^{(k)}$, $k = 0, 1, \ldots, (F-1)$.

C. Video Distortion Dynamics

The value of the distortion D at hop t along the path from the source to the destination node depends on the position of the first unrecoverable frame in the GOP. We define the process $C = \{C_t, \ t = 0, 1, \ldots\}$ such that C_t is the index of the first unrecoverable frame in the GOP structure at hop t. At each hop t the process takes values in the set:

$$\mathcal{C} = \{0, 1, 2, \ldots, F-1, F\} \quad (17)$$

The value 0 indicates that the first (I-frame) is lost and therefore the whole GOP is unrecoverable. A value between 1 and $(F-1)$ denotes that the corresponding P-frame is the first frame in the GOP that cannot be decoded correctly and the value F indicates that no frame has been lost thus far, yielding a distortion $D = 0$. The definition of the process C suggests that the sample paths of the process are non-increasing w.p.1. which means that $C_t \ge C_{t+1}$, for all t.

The dynamics of the process C and therefore of the video distortion D depend on the process M. The value of the process M at each hop t indicates the number of lost packets up to that point along the path from the source to the destination node. These losses specify the first unrecoverable frame in the GOP and hence, the value of the distortion D at that point on the path. The transition probabilities at hop $t = 0, 1, \ldots$, of the process C,

$$p_t(i, j) = P\{C_{t+1} = j \mid C_t = i\} \quad \text{for } i, j \in \mathcal{C} \quad (18)$$

specifying the likelihood that the first unrecoverable frame at hop $t + 1$ is j given that the first unrecoverable frame at hop t is i, can be computed using the distributions $\boldsymbol{\pi}_t^{(0)}, \boldsymbol{\pi}_t^{(1)}, \ldots, \boldsymbol{\pi}_t^{(F-1)}$ given by (15) and (16). In particular, we consider the following cases:

1) For $i = 0$: In this case the first unrecoverable frame at hop t is the first frame (I-frame) in the GOP. This means that the GOP is unrecoverable and the value of the process C for the rest of the path cannot be anything else other than 0. Therefore, the transition probabilities in this case are given by (19):

$$p_t(0, j) = \begin{cases} 1, & j = 0 \\ 0, & j = 1, 2, \ldots, F \end{cases} \quad (19)$$

2) For $i = 1, 2, \ldots, (F-1)$: When the first unrecoverable frame in the GOP at hop t is a P-frame, it is possible during the transition to the next $(t+1)$ hop to have packet losses that make an earlier frame in the GOP unrecoverable. This will happen if the number of lost packets in an earlier frame is such that, the total number of lost packets for the particular frame reaches the sensitivity of that frame type. This is used to compute the transition probabilities in (20).

3) For $i = F$: This corresponds to the case where no frames have been lost in the GOP up to hop t. The transition to the next hop may cause packet losses such that, either a frame in the GOP becomes unrecoverable or none is lost and no transition of C happens. The transition probabilities in this case are given by (21).

$$p_t(i,j) = \begin{cases} \pi_{t+1}^{(0)}(s_I), & j = 0 \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right) \prod_{k=1}^{j-1}\left(1 - \pi_{t+1}^{(k)}(s_P)\right) \pi_{t+1}^{(j)}(s_P), & j = 1, 2, \ldots, (i-1) \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right) \prod_{k=1}^{j-1}\left(1 - \pi_{t+1}^{(k)}(s_P)\right), & j = i \\ 0, & j = (i+1), \ldots, F \end{cases}$$

$$= \begin{cases} \pi_{t+1}^{(0)}(s_I), & j = 0 \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right)\left(1 - \pi_{t+1}^{(1)}(s_P)\right)^{j-1} \pi_{t+1}^{(1)}(s_P), & j = 1, 2, \ldots, (i-1) \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right)\left(1 - \pi_{t+1}^{(1)}(s_P)\right)^{j-1}, & j = i \\ 0, & j = (i+1), \ldots, F \end{cases} \qquad (20)$$

$$p_t(F,j) = \begin{cases} \pi_{t+1}^{(0)}(s_I), & j = 0 \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right) \prod_{k=1}^{j-1}\left(1 - \pi_{t+1}^{(k)}(s_P)\right) \pi_{t+1}^{(j)}(s_P), & j = 1, 2, \ldots, (F-1) \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right) \prod_{k=1}^{F-1}\left(1 - \pi_{t+1}^{(k)}(s_P)\right), & j = F \end{cases}$$

$$= \begin{cases} \pi_{t+1}^{(0)}(s_I), & j = 0 \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right)\left(1 - \pi_{t+1}^{(1)}(s_P)\right)^{j-1} \pi_{t+1}^{(1)}(s_P), & j = 1, 2, \ldots, (F-1) \\ \left(1 - \pi_{t+1}^{(0)}(s_I)\right)\left(1 - \pi_{t+1}^{(1)}(s_P)\right)^{F-1}, & j = F \end{cases} \qquad (21)$$

The value of the video transmission distortion D depends on the value of the process C at hop t. More specifically,

$$D_t = \begin{cases} 0, & \text{if } C_t = F \\ D^{(c)}, & \text{if } C_t = c \text{ and } 0 \leq c \leq (F-1), \end{cases} \qquad (22)$$

where $D^{(c)}$ is given by (4). Therefore, the dynamics of the video transmission distortion are defined by the transition probabilities $p_t(i,j)$ given by (19), (20) and (21).

IV. OPTIMAL ROUTING POLICY

Next, our objective is to find the path that yields the minimum video transmission distortion between any source and destination. By using the analysis presented in Section III we pose the problem as a stochastic optimal control problem where the control is the selection of the next node to be visited at each intermediate node from the source to the destination.

If $\mathcal{N} = \{1, 2, \ldots, N\}$ is the set of nodes in the network and $\mathcal{C} = \{0, 1, \ldots, F-1, F\}$ is the set of possible values for the process C described in Section III-C, we define the state space of our problem as:

$$\mathcal{X} = \mathcal{N} \times \mathcal{C}. \qquad (23)$$

Each state $\boldsymbol{x} \in \mathcal{X}$ is a tuple such that $\boldsymbol{x} = (n,c)$. The first component $n \in \mathcal{N}$ represents the current node on the path from the source to the destination. The second component $c \in \mathcal{C}$ points to the first unrecoverable frame in the GOP and therefore, specifies the video distortion at the current node.

Suppose that at the kth-hop of the path between the source and the destination the node is n_k. Suppose furthermore, that the first unrecoverable frame in the GOP structure is c_k. Then, the current state of the system is $\boldsymbol{x}_k = (n_k, c_k)$. At this point, the system needs to select the next node to be visited. Denote this selection by u_k. Clearly, the node u_k to be selected next

should belong to the set $\mathcal{U}(\boldsymbol{x}_k)$ of the one hop neighbors of n_k. This means that, if at stage k the state is $\boldsymbol{x}_k = (n_k, c_k)$ and a decision is made such that $u_k = u$, $u \in \mathcal{U}(\boldsymbol{x}_k)$ the new state at the next stage $k+1$ will be $\boldsymbol{x}_{k+1} = (u, c_{k+1})$. The selection of u as the next node specifies the packet loss probability β from the analysis in Section III-B and accounts for both channel induced and interference related failures. Moreover, it specifies the transition probabilities $p_k(c_k, c_{k+1})$ for the second component of the state. To make the dependence of these transition probabilities to the selection explicit, we use the notation $p_k(c_k, c_{k+1} \mid u)$.

We seek to find the optimal sequence of states $\boldsymbol{x}_0, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_T$ that minimizes the total video transmission distortion from the source to the destination node. The first component of each state that belongs to such an optimal sequence of states indicates the node that has to be visited next in the optimal path.

For the initial state $\boldsymbol{x}$ and the sequence of decisions $\boldsymbol{u} = (u_0, u_1, \ldots, u_{T-1})$, the cost to be minimized is defined as:

$$W(\boldsymbol{x}, \boldsymbol{u}) = \mathrm{E}_x \left[\sum_{k=0}^{T-1} g(\boldsymbol{x}_k, u_k) + k(\boldsymbol{x}_T) \right], \quad \boldsymbol{x} \in \mathcal{X}, \quad (24)$$

where T is the length of the path, the function $g(\cdot, \cdot)$ is the running cost and the function $k(\cdot)$ is the final cost. We call this optimization problem the *Minimum Distortion Routing (MDR)* problem.

The running cost is the video transmission distortion at state $\boldsymbol{x} = (n, c)$:

$$g(\boldsymbol{x}, u) = g((n,c), u) = \begin{cases} 0, & \text{if } c = F \\ D^{(c)}, & \text{if } 0 \leq c \leq (F-1). \end{cases} \qquad (25)$$

The final cost is defined to be:

$$k(\boldsymbol{x}) = k(n,c)$$

$$= \begin{cases} 0, & \text{if } n \text{ is the destination node} \\ & \text{and } c = F \\ D^{(c)}, & \text{if } n \text{ is the destination node} \\ & \text{and } 0 \le c < F \\ D^{\max}, & \text{otherwise.} \end{cases} \quad (26)$$

If s is the source and d is the destination of the connection, then the initial state $\boldsymbol{x}_0$ for the optimization problem is defined as $\boldsymbol{x}_0 = (s, F)$. Any state in the boundary set

$$\mathcal{B} = \{(d,c) \in \mathcal{X} \mid 0 \le c \le F\} \quad (27)$$

is a terminating state for the optimization problem.

If $\boldsymbol{u}^\star$ is an optimal decision sequence, we define the value function $J(\cdot)$ as

$$J(\boldsymbol{x}) = W(\boldsymbol{x}, \boldsymbol{u}^\star) = \min_{\boldsymbol{u}} W(\boldsymbol{x}, \boldsymbol{u}) \quad (28)$$

for an initial state $\boldsymbol{x} \in \mathcal{X}$. If at some stage k the state is $\boldsymbol{x}_k = (n_k, c_k)$, we define the minimum cost-to-go as:

$$J_k(\boldsymbol{x}_k) = \min_u \Bigg\{ g(\boldsymbol{x}_k, u) + \sum_{\boldsymbol{x}_{k+1}} p_k(c_k, c_{k+1} \mid u) J_{k+1}(\boldsymbol{x}_{k+1}) \Bigg\} \quad (29)$$

and for the final stage:

$$J_T(\boldsymbol{x}) = k(\boldsymbol{x}) \quad (30)$$

The MDR problem has the following properties:

Lemma 1. *MDR satisfies the overlapping property, i.e. the problem can be broken down into smaller problems that retain the same structure.*

Proof: From (29), it is clear that computing the cost-to-go $J_k(\cdot)$ requires the calculation of the cost-to-go $J_{k+1}(\cdot)$. This means, that the initial problem of finding the optimal route between a source and a destination node can be solved if the sub-problem of finding an optimal path between an intermediate node and the destination can be solved. ∎

Lemma 2. *MDR satisfies the optimal sub-structure property, i.e. the sub-path of an optimal path is optimal for the corresponding sub-problem.*

Proof: This is immediate from the definition of the cost-to-go function defined in (29). ∎

Theorem 1. *The MDR problem is solvable by dynamic programming.*

Proof: An optimization problem can be solved by dynamic programming if the problem satisfies both the overlapping and the optimal substructure properties [18]. The proof is immediate from Lemmas 1 and 2. ∎

Since the state space $\mathcal{X}$ is of finite dimension, the optimization problem can be solved via dynamic programming by back propagating the computation of the value of the cost-to-go function [19], [20] starting from the terminating states of the boundary set $\mathcal{B}$ and moving backwards towards the initial

state $\boldsymbol{x}_0$. If at some stage k the state is $\boldsymbol{x}_k = (n_k, c_k)$ we consider all possible neighbors n_{k+1} of node n_k that are one hop away. For each link (n_k, n_{k+1}) a packet loss probability β characterizes the quality of this specific link. Using this β we can compute the transition probability from the current state $\boldsymbol{x}_k = (n_k, c_k)$ to a new state $\boldsymbol{x}_{k+1} = (n_{k+1}, c_{k+1})$ through the probability $p_k(c_k, c_{k+1} \mid u = n_{k+1})$ that is defined in Section III-C for all possible values of the second component c of the state. Among the neighboring nodes of node n_k we choose as the next hop towards the destination node, the node that corresponds to the minimum cost-to-go at stage k defined in (29).

Discussion: In essence, the MDR routing policy, distributes the video frames (and the packets contained therein) across multiple paths and in particular minimizes the interference experienced by the frames that are at the beginning of a GOP (to minimize distortion). The I-frames are longer than other frames. Their loss impacts distortion more, and thus these are transmitted on relatively interference-free paths. The higher protection rendered to I-frames, is the key contributing factor in decreasing the distortion with MDR (we also observe this in both our simulations and testbed experiments).

V. Protocol Design

To compute the solution to the MDR problem described in Section IV, knowledge of the complete network (the nodes that are present in the network and the quality of the links between these nodes) is necessary. However, because of the dynamic nature and distributed operations of a network, such complete knowledge of the global state is not always available to the nodes. In practice, the solution to the MDR problem can be computed by the source node based on partial information regarding the global state that it gathers. The source node has to sample the network during a path discovery process in order to collect information regarding the state of the network.

The sampling process includes the estimation of the Expected Transmission Count (ETX) metric [3] for each wireless link in the network. These estimates provide a measure of the quality of the links. The estimation process can be implemented by tracking the successful broadcasting of probe messages in periodic time intervals. The ETX estimates computed locally in the neighborhood of a node are then appended in the `Route Request` messages during the *Route Discovery* phase. Upon reception of this message by the destination, a `Route Reply` message is sent back to the source that contains the computed ETX estimates, which are usable to compute β.

The source node then, can solve the optimization problem (Section IV) by using the information gathered via the sampling process described above. Specifically, upon receiving the `Route Reply` messages, the source node s follows the steps presented in Algorithm 1. It defines the initial state of the optimization problem as $\boldsymbol{x} = (s, F)$, where F is the GOP size. It defines the boundary set $\mathcal{B}$ which serves as the terminating set for the optimization process. Next, a call to Algorithm 2 produces the next node $u^\star$ in the path. Because of the stochastic nature of the second component c of the state, its next value has to be estimated. The estimation is based on the transition probabilities $p_t(c, c')$ given by (19), (20) and (21). In particular, the estimated value is the expected value

Algorithm 1 Path discovery (Uses Algorithm 2).

Input: source node s, destination node d
Input: frame size F
Output: route R from s to d
1: /* DSR *Route Discovery* Phase */
2: **send** Route Request
3: **receive** Route Reply(n_i, ETX_i) messages
4: $\mathcal{N} \triangleq \{\text{node-ids } n_i \text{ from Route Reply messages}\}$
5:
6: /* Path Discovery Initialization Phase */
7: $n \leftarrow s$
8: $c \leftarrow F$
9: $\mathcal{B} \triangleq \{(d, c) \mid 0 \leq c \leq F\}$
10: $R \leftarrow [\,]$
11: $\boldsymbol{x} \leftarrow (n, c)$
12: **append** $\boldsymbol{x}$ to R
13:
14: /* Path Computation */
15: **repeat**
16: $\quad u^\star \leftarrow$ Next_node_in_optimal path$(\boldsymbol{x}, \mathcal{B}, \mathcal{N})$
17: $\quad \hat{c} \leftarrow \text{E}[C_{\text{new}} \mid C_{\text{cur}} = c]$
18: $\quad n \leftarrow u^\star$
19: $\quad c \leftarrow \hat{c}$
20: $\quad \boldsymbol{x} \leftarrow (n, c)$
21: $\quad$ **append** $\boldsymbol{x}$ to R
22: $\quad \mathcal{N} \leftarrow \mathcal{N} - \{u^\star\}$
23: **until** $\boldsymbol{x} \in \mathcal{B}$

Algorithm 2 Next node in optimal path.

Input: initial state $\boldsymbol{x}_s$, boundary set $\mathcal{B}$
Input: set of available nodes $\mathcal{N}$
Input: frame size F
Output: next node $n^\star$ in the optimal path
1: /* Initialization Phase */
2: $\mathcal{C} \triangleq \{0, 1, \ldots, F\}$
3: $\mathcal{X} \triangleq \mathcal{N} \times \mathcal{C}$
4: $T \leftarrow \|\mathcal{X}\|$
5:
6: /* Optimal Control Computation */
7: **for** $i = T$ **to** 1 **do**
8: $\quad$ **if** $i = T$ **then**
9: $\quad\quad$ **for all** $\boldsymbol{x} \in \mathcal{X}$ **do**
10: $\quad\quad\quad J_i(\boldsymbol{x}) \leftarrow k(\boldsymbol{x})$
11: $\quad\quad$ **end for**
12: $\quad$ **else**
13: $\quad\quad$ **for all** $\boldsymbol{x} = (n, c) \in \mathcal{X}$ **do**
14: $\quad\quad\quad U(n) \leftarrow \{n' \mid n, n' \text{ 1-hop neighbors}\}$
15: $\quad\quad\quad j_i(\boldsymbol{x}, u) \leftarrow \{g(\boldsymbol{x}, u) + \sum_{\boldsymbol{x}'} p_i(c, c' \mid u) J_{i+1}(\boldsymbol{x}')\}$
16: $\quad\quad\quad J_i(\boldsymbol{x}) \leftarrow \min_{u \in U(n)} j_i(\boldsymbol{x}, u)$
17: $\quad\quad\quad P_i(\boldsymbol{x}) \leftarrow \arg\min_{u \in U(n)} j_i(\boldsymbol{x}, u)$
18: $\quad\quad$ **end for**
19: $\quad$ **end if**
20: **end for**
21: $n^\star \leftarrow P_1(\boldsymbol{x}_s)$
22: **return** $n^\star$

of the second component given its current value:

$$\hat{c} = \text{E}[C_{\text{new}} \mid C_{\text{cur}} = c]. \tag{31}$$

To avoid loops in the produced route, node $u^\star$ is removed from the set $\mathcal{N}$ of available nodes. The process is repeated with a new initial state $\boldsymbol{x} = (u^\star, \hat{c})$ until the boundary set $\mathcal{B}$ is reached. In each iteration, Algorithm 2 is called to determine the next node on the path from the source s to the destination d. Algorithm 2 takes as an input an initial state $\boldsymbol{x}$, a boundary set $\mathcal{B}$, the GOP size F and the set $\mathcal{N}$. It solves the dynamic programming problem described in Section IV by first creating the state space of the system and then using the value iteration method, starting from the boundary set and moving backwards. At each stage of the process it also computes the optimal policy. At the end of the computation, the ID of the best node to be selected is returned by using the optimal policy for the first stage.

In the source routing scheme the routing decisions are made at the source node ahead of time and before the packet enters the network. Therefore, source routing is an open loop control

	Set-I	Set-II	Set-III
GOP Size	5	10	10
Frames per second	30	30	15
Rate	273 kbps	273 kbps	136 kbps
Frame Size	QCIF (176x144)		
MTU	1024 bytes		

problem where all decisions have to be made in the beginning. The decisions are taken sequentially; a decision at a stage corresponds to the choice of the next hop node at the node corresponding to the stage. The source node cannot know exactly the state $\boldsymbol{x}_k = (n_k, c_k)$ at the kth stage of the selection process because of the randomness of the second component of the state. It has to estimate at each stage, the value of c_k and use this estimate to make a decision for that stage.

VI. RESULTS

We show the performance gains of the proposed routing scheme via extensive simulations and testbed experiments. For the simulation experiments we use the network simulator ns-2 [21]. The simulator provides a full protocol stack for a wireless multi-hop network based on IEEE 802.11. We extend the functionality of ns-2 by implementing our proposed routing scheme on top of the current protocol stack. For the testbed experiments we implement our scheme using the Click modular router [22], [23]

Further, we use EvalVid [24], which consists of a set of tools for the evaluation of the quality of video which is transmitted over a real or simulated network. The tool-set supports different performance metrics such as the Peak Signal to Noise Ratio (PSNR) and the Mean Opinion Score (MOS) [25]. To adapt the EvalVid to the ns-2 simulator we follow the procedure described in [26]. Specifically, for each simulated video flow between two nodes in the network we need to produce a sequence of files. We start with the initial uncompressed video file which consists of a sequence of YUV frames [27]. Using the EvalVid tool-set we transform the YUV format first to the MP4 and then to the MPEG4 format which contains hints of how the video file should be transmitted over a network. We then need to capture a log from an attempted transmission over a real network. This log indicates which frame and at what time instance was transmitted over the network. The log is fed as an input to the ns-2 simulation which plays back the video transmission producing at the end, two sets of statistics regarding the transmission, one for the sender and one for the receiver. By applying the EvalVid tool-set on this sequence of files we can reconstruct the video file as it is received by the destination and compare it to the initial video file. The comparison provides a measure of the video quality degradation due to the transmissions over the network.

A. Simulation Results

To evaluate the performance of the MDR protocol we compare it against the minimum ETX routing scheme. We consider a wireless multi-hop network that covers an area of $1000 \times 1000 \text{ m}^2$. The nodes are distributed over this area according to a Poisson random field. Each node uses the IEEE 802.11b protocol where the propagation model is the Two Ray Ground, yielding a communication range of about 250 m.

In Table I, three sets of values are defined for the video encoding parameters. We vary the Group of Pictures (GOP) size and the frame rate and thus effectively, the video encoding rate. We keep the frame size constant as per the QCIF standard (176 x 144 pixels) and set the maximum packet size to 1024 bytes. Our simulation experiments focus on three metrics: (i) the PSNR, which is an objective quality measure, (ii) the MOS, which is a subjective quality metric and (ii) the delay experienced by each video connection.

The effect of the node density on the PSNR is shown in Fig. 1. We plot the average PSNR for 5 and 10 concurrent video connections for different node densities and for Set-I of the video encoding parameters of Table I. We also plot the performance of our proposed scheme (MDR) when instead of estimating the per link packet loss probabilities through the ETX metric, we use the model in Section III-A to do so. In this case, we assume full knowledge of the network topology and so, the state space where we solve the optimal control problem of Section IV is a superset of the state space when we collect the local estimates of ETX through the network.

We then fix the number of nodes to 20 (distributed as described earlier) and compute the PSNR of each video connection when (i) the network serves 4 concurrent connections and (ii) when the number of concurrent connections is 8. In each case, the source–destination pairs are chosen uniformly from among the nodes in the network. We define the tail distribution of PSNR as the probability $P\{\text{PSNR} > x\}$ and plot it in Fig. 2 for the different traffic loads. The tail distribution of PSNR that corresponds to Set-II of the video encoding parameters is shown in Fig. 2(a). For both the light and heavy traffic loads (4 and 8 concurrent connections respectively), the MDR protocol performs better, providing a higher percentage of paths that have a given PSNR value. As expected, a performance degradation is observed for both schemes when the traffic load increases. This is due to the fact that under heavier traffic conditions in the network, the interference becomes more prevalent; furthermore, interference across adjacent links can be correlated in some cases. Under such network conditions, the benefits from the distortion-based optimization have a greater impact on the path selection process for the different types of frames in a video GOP as discussed earlier. The I-frames are sent on relatively uncongested paths. With 4 concurrent connections the median of PSNR is 17 for the minimum ETX policy and 18 for the MDR protocol. The median decreases when the traffic load increases and it is 9.5 and 10 for the minimum ETX and the application-aware schemes, respectively. The tail distribution of PSNR that corresponds to the parameters of Set-III is shown
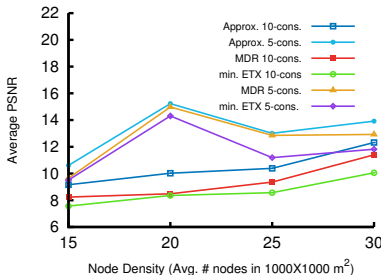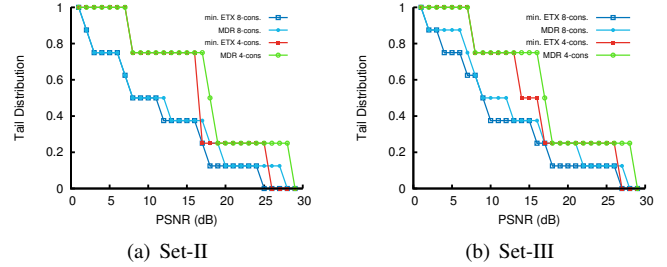

(a) Set-II   (b) Set-III

Fig. 2.   Tail distribution of PSNR.

in Fig. 2(b). As is the case for Set-II, a large GOP size results in a denser state space, and therefore a better performance for the MDR protocol. In the case of the light traffic loads (4 concurrent connections) the median for the PSNR is 15 for the minimum ETX scheme and 17 for MDR. Under heavier traffic loads (8 concurrent connections) the median for the PSNR is 9 for the minimum ETX scheme and 10.5 for the MDR protocol.

The effects of the sensitivities, s_I and s_P, on the MDR-protocol are shown in Fig. 3. As before, the number of randomly placed nodes is set to 20. We compare the performance of the MDR protocol when $(s_I, s_P) = (1, 1)$ and $(s_I, s_P) = (2, 4)$. In the first case, the sensitivity to the packet losses per frame is set to the maximum; in this case a single packet loss in a frame causes the frame to be unrecoverable. Fig. 3(a) and Fig. 3(b) present the same comparison for Set-II and Set-III of the encoding parameters. In both cases, relaxing the sensitivity of an I or P-frame to packet losses (i.e. increasing the value of s_I and s_P) deteriorates the performance of the scheme. A lower sensitivity (larger values of s_I and s_P) diminishes the impact of packet losses on the video distortion thus limiting the performance gains from using the scheme. For Set-II, the median of the PSNR is 17 for the minimum ETX scheme and 18 and 16 for MDR for $(s_I, s_P) = (1, 1)$ and $(s_I, s_P) = (2, 4)$, respectively. When the video encoding parameters of Set-III are used, the median values of the PSNR are 15 for the minimum ETX case and 17 and 15 for the MDR protocol when $(s_I, s_P) = (1, 1)$ and $(s_I, s_P) = (2, 4)$ respectively.

Although the PSNR is the most widespread objective metric to measure the digital video quality, it does not always capture user experience. A subjective quality measure that tries to capture human impression regarding the video quality is the Mean Opinion Score (MOS). The metric uses a scale from 1 (worst) to 5 (best) to represent user satisfaction when watching a video clip [25].

To evaluate the MOS with the MDR and ETX-based routing, we consider the wireless multi-hop network with the average number of nodes equal to 20 (distributed as discussed earlier). The initial raw video is processed using the H.264 encoder
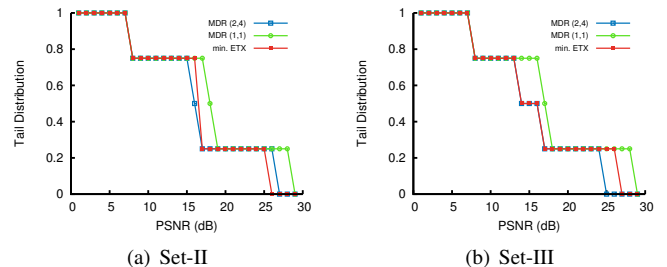

Fig. 1.   Average PSNR for 5 and 10 video connections (Set-I).

(a) Set-II   (b) Set-III

Fig. 3.   PSNR dependence on packet loss sensitivity.

Fig. 4. Average Mean Opinion Score.

(a) Mean delay.

(b) Variance of delay.

Fig. 5. Delay characteristics – 5 connections (Set-II).
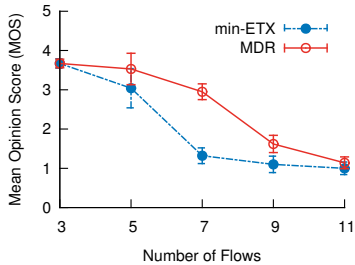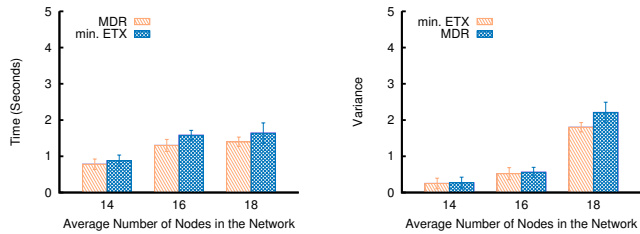
with a maximum GOP size of 30 frames and a sampling frequency of 30 frames per second. Fig. 4 shows the average MOS as the number of concurrent video flows in the network increases. When the number of connections is 3 the traffic load is low and so both the ETX-based routing and MDR provide similar user experience regarding video quality. As the traffic load increases the distortion-based routing distributes the load across the network causing the I-frames to avoid highly congested areas. This results in higher MOS values which translates to a better user experience.

The delay characteristics of the two routing schemes are shown in Fig. 5 for Set-II of the video encoding parameters. The nodes are again randomly distributed according to a Poisson random field with varying density with values 14, 16 and 18. The traffic load corresponds to 5 concurrent video connections. We compute and plot the mean and variance of the end-to-end delay for the 5 connections along with the 95% confidence intervals. As seen in Fig. 5, for all three different node densities, the MDR protocol produces routes that exhibit less variability compared to the routes computed by the minimum ETX scheme. Smaller variability implies less jitter, which in turn suggests a better video quality as perceived by the end user. Moreover, because of the smaller variability, the required sizes of buffers at the intermediate nodes is smaller. Note that this benefit is in addition in the reduction in distortion as discussed above. The primary reason for this reduction in the delay is that the distortion-aware approach tries to avoid paths which are congested; ETX on the other hand results in convergence of flows onto a few good paths. For both routing schemes, the mean and variance of the delay increase with the average number of nodes in the network. As the network becomes denser, the effect of interference becomes more profound increasing the number of retransmissions and thus, the delay. In contrast, a sparser network topology provides a smaller number of "good" routes, and thus, it is more difficult to separate flows and cope with congestion. It is in the moderate density regions, where the MDR protocol provides the most benefits in terms of delay and jitter.

B. Testbed Experiments

Next we evaluate the MDR protocol on a wireless indoor testbed comprised of 30 nodes[1]. The nodes are based on the Soekris net5501 hardware configuration, and run a Debian Linux distribution. Each node is equipped with 500 MHz CPU, 512 Mbytes of RAM, and a WN-CM9 wireless mini-PCI card, which carries the AR5213 Atheros main chip. Each node

[1]We omit a link pointing to a detailed description of our testbed deployment for purposes of maintaining anonymity.

is using IEEE 802.11a to avoid interference from co-located campus networks. To further minimize interference from these other networks, all experiments performed at night.

The experiment setup consists of an initial raw video processed using the H.264 encoder with a maximum GOP size of 30 frames. The traffic load ranges from 2 to 12 concurrent video flows, where the sender and receiver pairs are randomly selected. Each scenario is repeated 5 times.

To capture the effect of the ETX-based and MDR routing schemes on the user experience, we measure the average MOS as the number of concurrent video flows in the network increases. Fig. 6 shows that as the number of video connections in the network increases, the average MOS for both schemes decreases. However, when the traffic load increases, the MDR protocol computes multiple paths between the source and the destination nodes, and is better in distributing the load across the network such that the frames at the beginning of a GOP avoid congestion. On the other hand, the shorter paths computed through the ETX-based scheme become quickly congested resulting in heavy packet losses. As discussed, we observe that this primarily has a negative impact on correctly decoding the relatively longer (but more important) I-frames resulting in a worse user experience.

A visual comparison between Fig. 4 and Fig. 6 immediately shows the similarity in behaviors between our simulations and real experiments, thereby validates the realism of our simulations. Fig. 7 shows snapshots from video clips transmitted over the testbed under different traffic conditions for both the ETX-based and the MDR protocols. As shown in Fig. 6 when there are 2 connections in the network, the MOS for both routing schemes is the same. This is reflected in Fig. 7(a) and Fig. 7(b) where both snapshots are of very similar quality; in this case, the traffic load is fairly low and congestion is not a big issue (the flows do not cause high levels of interference to each other). When there are 8 concurrent video connections (and interference across connections is more prevalent), the MDR protocol achieves a higher MOS compared to the ETX-based scheme. This is visually depicted in Fig. 7(c) and Fig. 7(d) where the snapshot in the case of MDR is much more clear than the noisy snapshot form the ETX-based protocol. Specifically, our protocol distributes the I-frames across diverse paths with low interference; P-frames that are towards the end of GOPs are relatively packed together onto more congested paths. The ETX metric, which is agnostic to video semantics, does not distinguish between frames and packs them together, causing high distortion.

VII. CONCLUSION

In this paper, we argue that a routing policy that is application-aware is likely to provide benefits in terms of user-
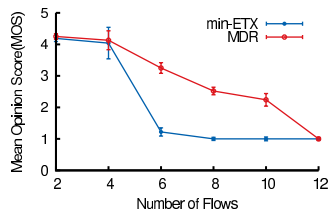
Fig. 6. Average value of Mean Opinion Score (MOS) for a different number of concurrent video flows.

(a) Video snapshot– MDR (2 connections).

(b) Video snapshot– ETX (2 connections).

(c) Video snapshot– MDR (8 connections).
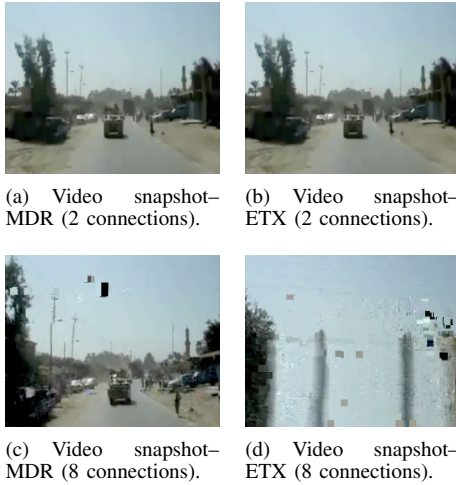
(d) Video snapshot– ETX (8 connections).

Fig. 7. User experience under different traffic loads.

perceived performance. Specifically, we consider a network which primarily carries video flows. We seek to understand the impact of routing on the end-to-end distortion of video flows. Towards this, we construct an analytical model which ties video distortion to the underlying packet loss probabilities. Using this model, we find the optimal route (in terms of distortion) between a source and a destination node using a dynamic programming approach. Unlike traditional metrics such as ETX, our approach takes into account correlation across packet losses which influence video distortion. Based on our approach, we design a practical routing scheme which we then evaluate via extensive simulations and testbed experiments. Our simulation study shows that the distortion (in terms of PSNR) is decreased by 20% compared to ETX-based routing. Moreover, the user experience degradation due to increased traffic load in the network, is kept to a minimum.

REFERENCES

[1] ISO/IEC JTC1/SC29/WG11, "ISO/IEC 14496 – Coding of audio-visual objects," http://mpeg.chiariglione.org/standards/mpeg-4/mpeg-4.htm.

[2] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.

[3] D. S. J. D. Couto, D. Aguayo, J. Bicket, and R. Morris, "A high-throughput path metric for multi-hop wireless routing," in *Proceedings of the 9th International Conference on Mobile Computing and Networking (MobiCom)*, San Diego, CA, Sept. 2003.

[4] J. M. Boyce, "Packet loss resilient transmission of MPEG video over the internet," *Elsevier Signal Processing: Image Communication*, vol. 15, no. 1-2, pp. 7–24, Sept. 1999.

[5] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Real-time communications over unreliable networks," *IEEE Signal Process. Mag.*, vol. 17, no. 4, pp. 61–82, July 2000.

[6] R. Zhang, S. L. Regunathan, and K. Rose, "Video coding with optimal inter/intra-mode switching for packet loss resilience," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 966–976, June 2000.

[7] J. Xiao, T. Tillo, and Y. Zhao, "Error-resilient video coding with end-to-end rate-distortion optimized at macroblock level," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, 2011.

[8] M. T. Ivrlač, L. U. Choi, E. Steinbach, and J. A. Nossek, "Models and analysis of streaming video transmission over wireless fading channels," *Elsevier Signal Processing: Image Communication*, vol. 24, no. 8, pp. 651–665, Sept. 2009.

[9] Y. J. Liang, J. G. Apostolopoulos, and B. Girod, "Analysis of packet loss for compressed video: Effect of burst losses and correlation between error frames," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 7, pp. 861–874, July 2008.

[10] D. Li and J. Pan, "Performance evaluation of video streaming over multi-hop wireless networks," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 338–347, Jan. 2010.

[11] Y. Wang, Z. Wu, and J. M. Boyce, "Modeling of transmission-loss-induced distortion in decoded video," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 6, pp. 716–732, June 2006.

[12] D. Migliorini, E. Mingozzi, and C. Vallati, "Performance evaluation of H.264/SVC video streaming over mobile WiMAX," *Elsevier Computer Networks*, vol. 55, no. 15, pp. 3578–3591, Oct. 2011.

[13] B. Rong, Y. Qian, K. Lu, R. Qingyang, and M. Kadoch, "Multipath routing over wireless mesh networks for multiple description video transmission," *IEEE J. Sel. Areas Commun.*, vol. 28, no. 3, pp. 321–331, Apr. 2010.

[14] M. M. Hira, F. A. Tobagi, and K. Medepalli, "Throughput analysis of a path in an IEEE 802.11 multihop wireless network," in *Proceedings of the IEEE Wireless Communications and Networking Conference, (WCNC 2007)*, Hong Kong, China, Mar. 2007.

[15] M. Garetto, T. Salonidis, and E. W. Knightly, "Modeling per-flow throughput and capturing starvation in CSMA multi-hop wireless networks," in *Proceedings of the IEEE Conference on Computer Communications, (INFOCOM 2006)*, Barcelona, Spain, Apr. 2006.

[16] K. Medepalli and F. A. Tobagi, "Towards performance modeling of IEEE 802.11 based wireless networks: A unified framework and its applications," in *Proceedings of the IEEE Conference on Computer Communications, (INFOCOM 2006)*, Barcelona, Spain, Apr. 2006.

[17] J. S. Baras, V. Tabatabaee, G. Papageorgiou, and N. Rentz, "Performance metric sensitivity computation for optimization and trade-off analysis in wireless networks," in *Proceedings of the IEEE Global Communications Conference (IEEE GLOBECOM 2008)*, New Orleans, LA, Nov. 2008.

[18] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*. McGrawHill, 2000.

[19] D. P. Bertsekas, *Dynamic Programming and Optimal Control, Vol I*, 2nd ed., ser. Optimization and Computation Series. Belmont, MA: Athena Scientific, 2000.

[20] M. L. Puterman, *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley-Interscience, 2005.

[21] The Network Simulator, "ns-2," http://nsnam.isi.edu/nsnam/index.php/Main_Page.

[22] "The Click Modular Router Project," http://www.read.cs.ucla.edu/click.

[23] E. Kohler, R. Morris, B. Chen, J. Jannotti, and M. F. Kaashoek, "The Click modular router," *ACM Transactions on Computer Systems*, vol. 18, no. 3, pp. 263–297, Aug. 2000.

[24] J. Klaue, B. Rathke, and A. Wolisz, "EvalVid–A framework for video transmission and quality evaluation," in *Proceedings of the 13th International Conference on Modelling Techniques and Tools for Computer Performance Evaluation*, Urbana, IL, Sept. 2003.

[25] A. Bovik, *The Essential Guide to Video Processing*. Academic Press, 2009.

[26] C.-H. Ke, C.-K. Shieh, W.-S. Hwang, and A. Ziviani, "An evaluation framework for more realistic simulations of MPEG video transmission," *Journal of Information Science and Engineering*, vol. 24, no. 2, pp. 425–440, Mar. 2008.

[27] C. A. Poynton, *A Technical Introduction to Digital Video*. Wiley, 1996.