Fine-Granularity Transmission Distortion Modeling for Video Packet Scheduling Over Mesh Networks

Yongfei Zhang, Shiyin Qin, and Zhihai He, *Senior Member, IEEE*

***Abstract*—Packet scheduling is a critical component in multi-session video streaming over mesh networks. Different video packets have different levels of contribution to the overall video presentation quality at the receiver side. In this work, we develop a fine-granularity transmission distortion model for the encoder to predict the quality degradation of decoded videos caused by lost video packets. Based on this packet-level transmission distortion model, we propose a content-and-deadline-aware scheduling (CDAS) scheme for multi-session video streaming over multi-hop mesh networks, where content priority, queuing delays, and dynamic network transmission conditions are jointly considered for each video packet. Our extensive experimental results demonstrate that the proposed transmission distortion model and the CDAS scheme significantly improve the performance of multi-session video streaming over mesh networks.**

***Index Terms*—Delay constraints, dynamic network, priority scheduling, transmission distortion model, video streaming.**

I. Introduction

WITH recent technological advances in video compression and networking, multi-session video transmission over mesh networks gains increasingly research interest and enables a variety of multimedia applications, such as video on demand and online video chatting over community networks and Internet. Video streaming over mesh networks is a challenging task because compressed videos are very sensitive to transmission errors (e.g., packet loss), video transmission has stringent delay requirements, and video sessions compete with each other for limited network resources to maximize their quality-of-service [1], [2].

A. Related Work

In this work, we consider the problem of packet scheduling for multi-session video transmission over mesh networks. This problem has been studied in different scenarios, such as over Internet, IEEE 802.11 wireless LAN, etc. [1], [2]. A number of scheduling algorithms have been developed in the literature for transmitting packets from generic data streams, such as audio and data files. They aim to maximize the quality-of-service which is often measured by average throughput, end-to-end delay or packet loss rate [3]–[5]. Unlike other types of data, videos are non-stationary data encoded with motion prediction and hybrid coding schemes. Different video packets have different levels of contribution to the overall video presentation quality at the receiver side. Motivated by this fact, a number of content-aware packet scheduling methods have been developed [7]–[19]. The key in content-aware scheduling is to determine the importance level of each video packet/frame. For embedded video encoders, such as wavelet-based video encoding schemes [6], packets in the video stream are naturally organized according to their importance levels in a hierarchical manner. van der Schaar *et al.* have shown that partitioning an embedded video stream into several priority classes (or quality layers) can improve the overall received video quality [7]. Based on this idea, a distributed cross-layer approach using priority queuing is proposed in [2] for multi-session video streaming over multi-hop wireless networks. An exponential model has been used to estimate the packet priority for wavelet-based video encoders [8]. Miao and Ortega uses a performance metric called expected run-time distortion to determine the importance of each packet [9]. This low-complexity heuristic scheduling algorithm shows a very promising performance [10].

Note that, in practice, most video encoding and communication systems are based on hybrid video encoders, such as H.264 and MPEG-4 [11]. Therefore, it becomes an important task to develop efficient content-aware packet scheduling methods for hybrid video encoding. In [12], a heuristic approach is proposed to determine the importance levels of video frames based on frame types (I, P, or B frames) and positions in a group of pictures (GOPs). In [13], the video packet's relative position within the GOP and its motion-texture context are used. A content-aware resource allocation and packet scheduling scheme is developed by Pahalawatta *et al.* [14], where the distortion is defined as the sum of absolute pixel difference between the decoded frames and the error-free reconstruction and then a distortion-based utility function is adopted for the gradient-based scheduling policy. Politis [15] uses a recursive formula to predict frame-level transmission distortion for three different error patterns, namely, isolated errors, burst errors, and errors with lags, where offline measurements are required for model parameter estimation. A rate-distortion optimized packet scheduling scheme is proposed by Chakareski and Frossard in [16]. Packet scheduling of video frames during network congestion

Manuscript received April 09, 2009; revised September 14, 2009. First published November 13, 2009; current version published December 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. James E. Fowler.

Y. Zhang was with the Department of Electrical and Computer Engineering, University of Missouri. He is now with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China (e-mail: zhangyf.ac@gmail.com).

S. Qin is with the School of Automation Science and Electrical Engineering, Beihang University, Beijing 100191, China (e-mail: qsy@buaa.edu.cn).

Z. He is with the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211 USA (e-mail: HeZhi@missouri.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TMM.2009.2036290

and poor channel conditions has been studied in [17], where the importance level of each frame, transmission deadline, and the frame size are jointly considered for deciding frame drops. Congestion-distortion optimized scheduling (CoDiO) proposed by Setton and Girod [18] aims to minimize a Lagrangian cost $D + \lambda\Delta$ where D is the distortion of decoded video and Δ is the expected end-to-end delay used as a congestion metric.

While a number of heuristic models have been developed in the literature for characterizing the importance levels of video packets, there is a lack of a concrete packet-level transmission distortion model for highly efficient content-aware packet scheduling. Furthermore, the accuracy and reliability of packet-level transmission distortion modeling and their impact on the overall packet scheduling performance have not been adequately studied.

B. This Work and Major Contributions

We recognize that the ultimate goal in video transmission is to maximize the video quality at the decoder side. Therefore, the importance level of each video packet should be determined by the amount of video quality degradation in the decoded videos caused by the loss of this video packet. In this paper, we provide a further understanding of the propagation behavior of transmission errors at fine scales, i.e., MB and packet-level rather than frame or sequence-level as in the literature [18], [22], [26]. We then establish a fine granularity transmission distortion model for accurate and fast prediction of video quality degradation caused by lost MBs or video packets at the encoder side before video transmission. The performance of the proposed model is extensively evaluated. Based on this model, we develop a content-and-deadline-aware scheduling scheme for multi-session video streaming over multi-hop mesh networks, which jointly considers content priority of video packets, queuing delays, and dynamic network conditions. Our extensive experimental results demonstrate that the proposed transmission distortion model and the CDAS scheme significantly improve the performance of multi-session video transmission over mesh networks.

C. Paper Organization

The rest of the paper is organized as follows. Section II presents our system specifications, models and assumptions that we use in this work. Section III presents our fine-granularity packet-level transmission distortion model. The proposed CDAS scheme is presented in Section IV. The performances of the proposed packet-level transmission distortion model and CDAS scheme are evaluated in Section V, and Section VI concludes the paper.

II. System Specifications

In this paper, we consider multi-session video streaming over mesh networks. A mesh network can be represented by a triplet $\boldsymbol{T}\{\boldsymbol{G}, \boldsymbol{L}, \boldsymbol{C}\}$, where $\boldsymbol{G}$ defines the network nodes and $\boldsymbol{L}$ denotes the available transmission links in the multi-hop wireless network, while the interference matrix $\boldsymbol{C}$ indicates whether two different links can transmit data simultaneously [4]. We assume that the ith session consists of H_i hops, and at each intermediate node reside a transmission buffer and a packet scheduler. As illustrated in Fig. 1, the intermediate and destination nodes receive packets from incoming links and store them in the buffer. The buffer size is empirically set to a fixed size, e.g., 1 M byte. When the buffer level exceeds the buffer size, the packets with the least overall packet scheduling priorities in the buffer are choose to be dropped. The packet scheduler decides at each time slot which packet in the buffer to transmit or drop according to our proposed scheduling policy. We assume that a predetermined transmission opportunity interval is reserved for each video session on the network nodes along its transmission path. This type of reservation mechanism can be implemented and controlled either by a central coordinator in the multi-hop network [19] or by a distributed solution using an overlay network infrastructure [20].

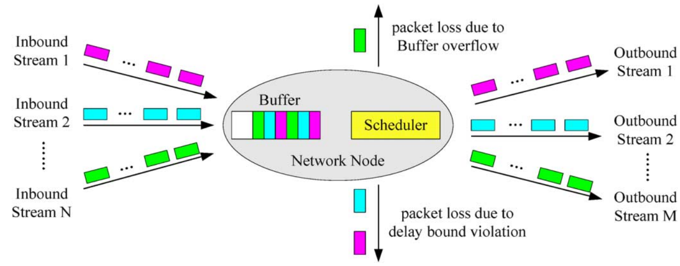

Fig. 1. Illustration of packet scheduling at an intermediate node of a mesh network.

We also assume that adaptive modulation and error resilient mechanisms (e.g., FEC and/or ARQ) are employed to ensure that each packet is correctly received or the transmission bit error rate (BER) is kept very low and negligible [20]. In this case, the guaranteed transmission bandwidth for each video session on each link represents the number of information bits per second that can be successfully transmitted. In this case, the major cause of transmission distortion is packet loss due to buffer overflow and delay bound violation [11]. Once a packet is lost, it will cause distortion in the decoded video frame. More importantly, due to motion compensation, the decoding error will propagate along the motion prediction path, accumulate over time, and significantly degrade the overall video presentation quality. This type of picture distortion caused by packet loss is referred to as *transmission distortion* [22].

III. Analysis and Modeling of Packet-Level Transmission Distortion

Recently, a number of methods and algorithms have been developed in the literature for transmission distortion analysis and modeling [17], [22]–[27], which can be classified into two major categories. In the first category, mathematical models are established to predict the transmission distortion. Existing distortion models have been focusing on transmission distortion modeling and prediction at the frame or sequence levels [17], [22], [23], [25]. This usually involves offline measurement of transmission distortion to estimate the model parameters, which is however not allowed in real-time video transmission. Besides, it does not explicitly consider the input video characteristics or the encoding structure and parameters, and hence, the obtained model cannot be generalized and these model parameters need to be obtained by curve fitting for each sequence. What is more, these frame or sequence-level distortion model cannot provide the corresponding distortion for each specific packet, which is however required for many applications like distortion-optimized packet scheduling. In the second category, the video encoder attempts to simulate the decoding behavior in a statistical manner so as to estimate the transmission distortion at the pixel level [26]. This often involves high computational complexity and implementation cost. In addition, it does not lead to an analytic model for transmission distortion, which is however needed for distortion-optimized applications as performance optimization and resource allocation [22].

In this work, we aim to develop a general and accurate packet-level transmission distortion model, which will allow us to predict the transmission distortion for each packet of different video sequences without online/offline training or parameter fitting. This model works for different sequences with very low computational complexity, and thus enables real-time packet scheduling and performance optimization.

It is worth noting that each packet consists of a number of macro blocks (MBs). This number may change dramatically from one packet to another, depending on the specific packetization scheme, as well as the coding complexity of the corresponding picture region. Therefore, to construct an efficient and flexible scheme for packet-level transmission distortion estimation, we need to model and estimate the transmission distortion at the MB level. There are two major issues that need to be carefully studied. The first issue is how to accurately model and estimate the transmission distortion of each MB. The second one is how to derive the packet-level transmission distortion from the estimation results at the MB level. In the following sections, we will present our approaches to address these issues.

Throughout the paper, unless otherwise specified, video sequences are encoded with MPEG-4 [11] with a quantization parameter (QP) of 8, a GOP size of $S = 15$ (an I frame followed by 14 P frames), and an intra refresh ratio (the fraction of intra-coded MBs) of 10%. When an MB is lost, we use a typical error concealment scheme, copying the MB data at the same position in the previous reconstructed frame. The distortion is measured in mean squared error (MSE) and the video quality in peak signal-to-noise ratio (PSNR).

A. Analysis and Modeling of MB-Level Transmission Distortion

In our previous work [22], we have developed a control system approach for frame-level transmission distortion modeling, where the transmission distortion $D_t(n)$ is modeled as the system response to an impulse input $\varepsilon(n)$, as shown in the following equation:

$$D_t(n) = \sum_{k=0}^{\infty} \varepsilon(k-n) H(k) \quad (1)$$

$$H(n) = D_t(n_0) e^{-\alpha(n)(n-n_0)}, \quad \alpha(n) = \frac{k_0}{M(n)} + k_1 \quad (2)$$

where the instantaneous transmission distortion $D_t(n_0)$ is the distortion caused at the frame where the transmission errors $\varepsilon(n)$ are introduced. We observed that the transmission distortion exhibit a fading behavior, as shown in (2). The fading factor $\alpha(n)$ depends on the motion reference ratio (MRR) $M(n)$, as shown in (2), where k_0 and k_1 are two constant parameters.

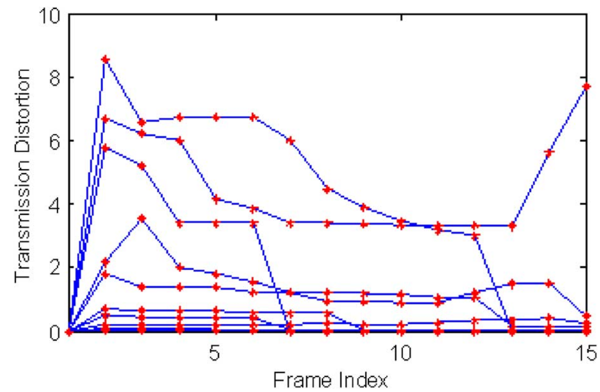

Fig. 2. Propagation behavior of MB-level transmission distortion.

In this work, we observe that this fading behavior of frame-level transmission distortion does not hold at the MB level. To demonstrate this, in the following experiment, we simulate MB losses at Frame 2 of QCIF (176 × 144) *Foreman* sequence. At the decoder side, we measure the corresponding transmission distortion in the subsequent frames (Frame 2 to 15). Fig. 2 shows the propagation behavior of several MBs. It can be seen that the nice fading behavior at the frame level does not hold at the MB level. This is because, compared to a video frame, an MB has a very limited amount of data. Furthermore, different MBs have a wide range of source characteristics. The propagation behavior of transmission distortion at the MB level is highly dependent on the specific fine-granularity characteristics of the MB, which makes the MB-level transmission distortion modeling more challenging than that at the frame level.

In this work, to accurately model the transmission distortion at the MB level, we extend the 2-D motion reference map (MRM) proposed in our previous work [22] into a 3-D motion reference map to capture the error propagation behavior at a finer scale. We find that there is a strong linear correlation between the MB-level transmission distortion and the motion reference ratio. Based on this finding, we develop a simple yet accurate MB-level transmission distortion model.

One major limitation of the 2-D motion reference map is that it only records whether a pixel is used as reference for motion prediction of pixels in the next frame or not. Thus it does not indicate how many times it is used as reference. Even though this is sufficient for frame-level transmission distortion modeling, it fails to capture the fine-granularity transmission distortion behavior at the MB level. As illustrated in Fig. 3, one pixel in Frame n might be used as reference by two pixels in Frame $n+1$. These two pixels might be used by three pixels in Frame $n+2$. If the pixel in Frame n is corrupted by transmission errors, all of these pixels in subsequent frames that directly or indirectly use this pixel as motion prediction reference will be affected. This error propagation will continue within the current GOP until the next intra frame.

Let $M_{i,j}(n)$ be the total number that pixel (i, j) in Frame n is directly or indirectly used as motion prediction reference in the

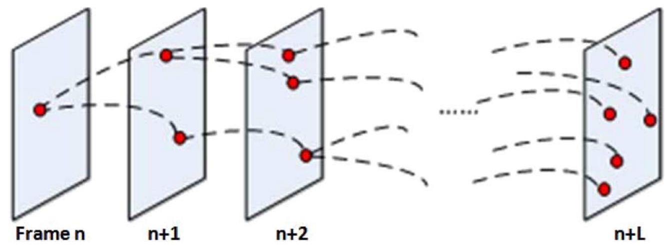

Fig. 3. Illustration of 3-D motion reference map.

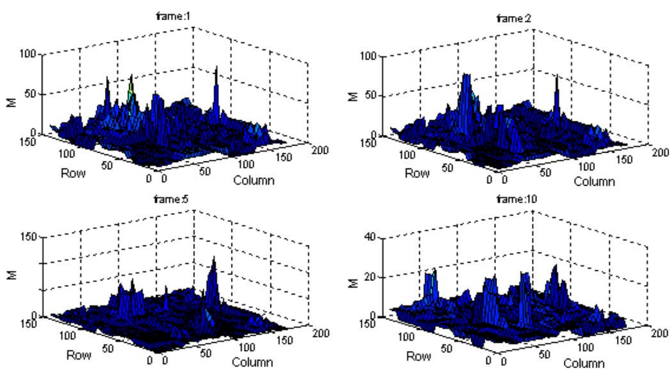

Fig. 4. 3-D motion reference maps of *Foreman* QCIF sequence.

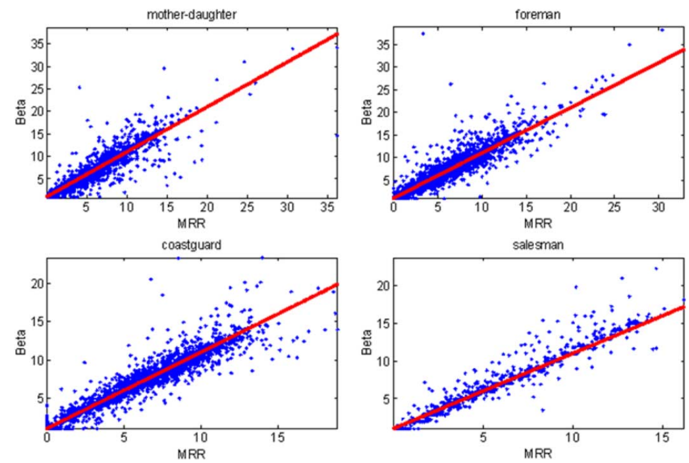

Fig. 5. Correlation between MRR and β for each MB.

TABLE I
CORRELATION COEFFICIENT BETWEEN MRR AND β

Test Video	Correlation	Test	Correlation
Mother&Daughter	0.9308	Salesman	0.9761
Carphone	0.9263	Akiyo	0.9806
Foreman	0.9462	News	0.9440
Coastguard	0.9693	Table	0.9071

subsequent frames. We refer to $M_{i,j}(n)$ as the motion reference map of Frame n. Note that $M_{i,j}(n)$ can be easily computed at the encoder after motion estimation. Fig. 4 shows the 3-D motion reference map for Frames 1, 2, 5, and 10 of the *Foreman* QCIF sequence. It can be seen that the MRM values change significantly from frame to frame and from pixel to pixel. Based on this MRM, we define the motion reference ratio $\mathcal{M}(n,m)$ for the mth MB in frame n as follows:

$$\mathcal{M}(n,m) = \frac{1}{K} \sum_{(i,j)\in MB(n,m)} M_{i,j}(n) \qquad (3)$$

where K is the total number of pixels in the MB. For example, K equals to 256 for a 16 $\times$ 16 MB.

In this work, we find that the transmission distortion of a lost MB has a strong correlation with its MRR. In order to demonstrate this, we perform the following experiments with eight QCIF size test video sequences: *Mother&Daughter*, *Carphone*, *Foreman*, *Coastguard*, *Salesman*, *Akiyo*, *News*, and *Tabletennis*, which have a wide range of scene activity characteristics.

Once an MB is lost, it will cause transmission distortion in the reconstructed frame to which the lost MB belongs. This transmission is referred to as the instantaneous transmission distortion D_0 as in [22], which can be obtained at the encoder since the encoder has access to the reconstructed frames. Let $\widehat{F}(n-1,m)$ and $\widehat{F}(n,m)$ be mth MB in the reconstructed frames $n-1$ and n at the encoder side. The instantaneous transmission distortion of the mth MB of Frame n $D_0(n,m)$ can be computed as follows:

$$D_0(n,m) = E\left[\widehat{F}(n-1,m) - \widehat{F}(n,m)\right]^2 \qquad (4)$$

where $E[\cdot]^2$ represents the MSE between two frames.

Besides, the loss of an MB might also result in distortions in subsequent frames within the current GOP due to error propagation. Let $\{\widehat{F}(n,m), \widehat{F}(n+1,m), \cdots, \widehat{F}(S,m)\}$ and $\{\breve{F}(n,m), \breve{F}(n+1,m), \cdots, \breve{F}(S,m)\}$ be the reconstructed n to Sth frames without and with the loss of the mth MB separately. Then, the overall transmission distortion caused by the loss of the mth MB in the nth frame is given by

$$D_t(n,m) = \sum_{i=n}^{S} E\left[\widehat{F}(i,m) - \breve{F}(i,m)\right]^2. \qquad (5)$$

To obtain the actual transmission distortion caused by the loss of an MB, we simulate this MB loss (with other MBs and frames to be error-free) within actual video encoding and decoding framework and measure the decoded video distortion. We then calculate the total accumulated video distortion (transmission distortion) caused by this lost MB as in (5). Define

$$\beta(n,m) = D_t(n,m)/D_0(n,m). \qquad (6)$$

In Fig. 5, we plot the value of β against the MRR for each MB for four test video sequences, where MBs of the first 45 frames (three GOPs) of each video sequence are considered. We can see that there is a very strong correlation between them. Table I shows the corresponding correlation coefficients for all eight test videos, which are very close to 1.0. Our simulations over other video sequences yield similar results.

This suggests a linear model for MB-level transmission distortion:

$$\beta(n,m) = a \cdot \mathcal{M}(n,m) + b \qquad (7)$$

i.e.,

$$D_t(n,m) = D_0(n,m) \times [a \cdot \mathcal{M}(n,m) + b] \qquad (8)$$

where a and b are two model parameters.

TABLE II
VALUES OF PARAMETERS a

Test Video	a	Video	a
Mother & Daughter	0.9935	Salesman	1.0365
Carphone	0.9803	Akiyo	0.9932
Foreman	1.0160	News	1.0368
Coastguard	0.9941	Table tennis	1.0142

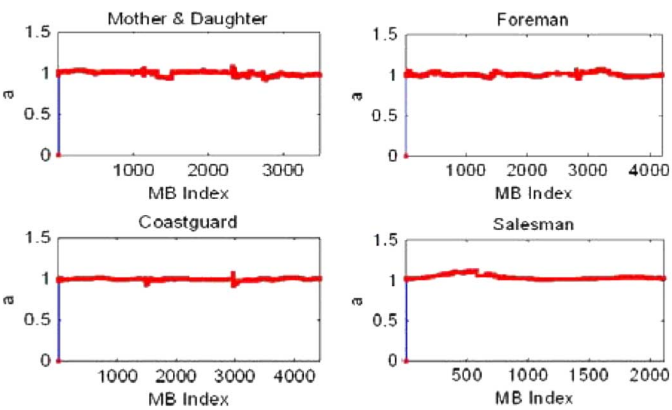

Fig. 6. Values of a for each MB of four test video sequences.

To successfully apply this model, we need to determine the values of $D_0, \mathcal{M}$, a and b. Note that $D_0(n, m)$ and $\mathcal{M}(n, m)$ can be both obtained at the encoder using (4) and (3), respectively. The estimation of these two model parameters a and b will be discussed in the next section.

B. Estimating the Model Parameters a and b for MB-Level Transmission Distortion

In the following, we discuss how to estimate the model parameters a and b in the transmission distortion model of (8). Considering a special case of the MBs in the last frame of a GOP, the pixels are not used as reference, so the motion reference ratio $\mathcal{M} = 0$; meanwhile, since no pixels are used as reference by subsequent frames, the error occurs in this frame will not propagate to other frames, which means $D_t = D_0$. Based on this fact, we have $b = 1$.

With the value of b obtained, only a is left to be determined. Table II shows the values of a for the first 45 frames (three GOPs) of all eight test sequences obtained using linear fitting and Fig. 6 shows the values of a for each MB of six video sequences: *Mother&Daughter*, *Foreman*, *Coastguard*, *Salesman*, *Akiyo*, and *Table Tennis*. We can see that, for all MBs of all sequences, the values of a are all very close to 1. Therefore, in the paper, we choose both parameters a and b in the proposed transmission distortions model to be 1. This significantly reduces the computational complexity and thus makes the model more practical for real-time applications. Our extensive experimental results will show that this simplified model also achieves very accurate transmission distortion estimation.

C. Deriving the Packet-Level Transmission Distortion

Using the linear model in (8), we can estimate the transmission distortion for each MB. Since a video packet consists of a group of MBs, the question now becomes: *is the transmission distortion of a lost video packet equals to the sum of transmission distortion values of all MBs within the packet*? Or, equivalently, *are transmission errors of different MBs independent of each other and is the transmission distortion additive*? In this section, we experimentally demonstrate that the MB-level transmission distortion has an additive property and the transmission distortion of a lost video packet can be computed by adding the transmission distortion values of all MBs within the packet. In Fig. 7, we plot the transmission distortion of lost video packets (in square-lines) for four test video sequences: *Salesman*, *Akiyo*, *News*, and *Table tennis*. Here, we only show the distortion estimation results for the first 45 frames (three GOPs) of each video sequence due to the limited spaces. However, the experimental results over other frames yield similar results. We also plot (in red circle-lines) the total transmission distortion of all MBs within the packet. Table III shows the correlation between the true packet-level transmission distortion and the estimated one based on the additive MB-level model for eight test videos. We can see that they are very close to each other. This suggests that the additive assumption of the MB-level transmission distortion does hold.

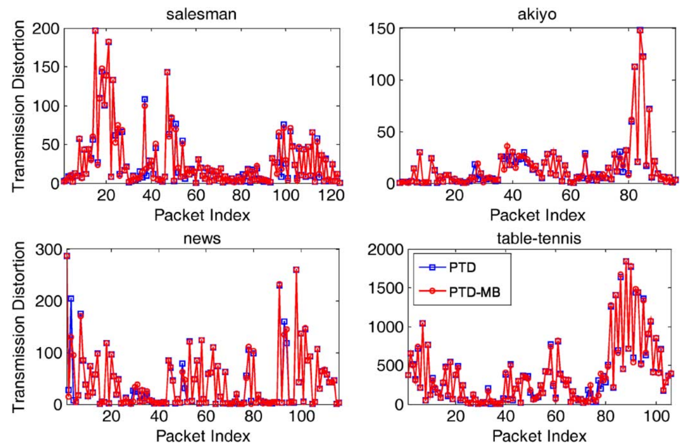

Fig. 7. Transmission distortion of lost video packets (PTD) versus estimation results based on the additive MB-level distortion model (PTD-MB).

TABLE III
CORRELATION COEFFICIENT BETWEEN PTD AND PTD-MB

Video	Correlation Coefficient	Video	Correlation Coefficient
Mother&Daughter	0.94	Salesman	0.98
Carphone	0.99	Akiyo	0.99
Foreman	0.96	News	0.98
Coastguard	0.98	Table tennis	0.98

D. Summary of Packet-Level Transmission Distortion Estimation Algorithm

Based on the analysis and results in the previous sections, we develop a simple yet accurate packet-level transmission distortion model, which operates at the encoder side to estimate the transmission distortion caused by lost video packets. The algorithm has the following major steps:

Step 1) Compute the motion reference ratio $\mathcal{M}$ using (3) and the instantaneous transmission distortion D_0 using (4) for each encoded MB during encoding process;

Step 2) Calculate the transmission distortion of each MB according to the linear model in (8);

Step 3) Compute the packet-level transmission distortion by adding the transmission distortion values of all MBs in the packet.

It can be seen that the proposed algorithm has very low computational complexity. The major computation is to obtain the motion reference ratio $\mathcal{M}$, which counts the number of times a pixel is directly or indirectly used as reference for motion prediction. Given the motion vector after motion search, only one addition operation is involved for each pixel if it is referenced by another pixel. Thus, the computational complexity increases linearly with the picture resolution. With this model, for each video packet, the transmission distortion can be computed at the encoder side even before transmission. We then normalize these values for packets of each video sequence to [0, 1], encode them with a binary representation, encapsulate into the packet headers and transmit them with the video packets. Then two bytes are sufficient for our applications, with high enough precision. Compared to the packet size (e.g., 500 bytes), this overhead is very small. The performance of the proposed packet-level transmission distortion model will be extensively evaluated in Section V.

IV. Content-Aware Packet Scheduling Under Delay Constraints

In this section, we present our proposed content-and-deadline-aware packet scheduling scheme for multi-session video streaming over mesh networks. In our scheme, we explicitly consider the content and the stringent delay constraint for each packet through a novel packet priority model, which consists of two major components: *content priority* and *network transmission priority*. The content priority is based on the packet-level transmission distortion model developed in the previous section. For the network transmission priority, we consider delay requirement and dynamic conditions of subsequent links [33]. Then these two priorities are combined into one metric, called *overall scheduling priority*, for each packet. The expected run-time distortion-based scheduling (ERDBS) scheme [9], [10] and the CoDiO [18] scheme are the two approaches most closely related to ours. In ERDBS, expected run-time distortion is used to evaluate the importance of each packet, where the distortion of each frame or layer is weighted by an urgency factor defined as a function of the playback deadline and round trip time (RTT). The CoDiO scheme formulates the scheduling problem under delay constraints as to minimize the expected Lagrangian cost $D+\lambda\Delta$, where D is the distortion of the received video stream and Δ is the expected end-to-end delay. In the following, we explain our proposed CDAS scheme in more detail.

A. Content Priority

The content priority of a video packet should be higher if the loss of this packet will cause a higher transmission distortion at the decoder. Therefore, a natural way to define the content priority is to use its transmission distortion [9], [14]–[28]. We observe that different video sequences have different characteristics and exhibit distinctive transmission distortion behaviors. For example, the values of transmission distortion for video packets in high-motion videos (e.g., *Foreman*) are often larger than those in low-motion videos (e.g., *Akiyo*), as shown in Fig. 8. If we use the transmission distortion as the measure of a packet's content priority directly, those video packets from low-motion video will most likely have low priority than those from high-motion videos and will not be scheduled for transmission during multi-session video streaming. This leads to a fairness issue. To address this issue, we propose to use the normalized transmission distortion as the measure of content priority P_c:

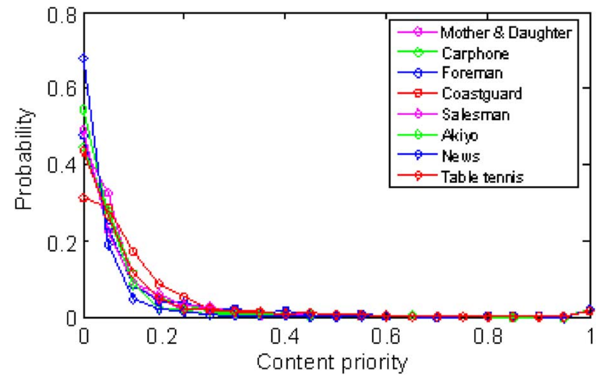

Fig. 8. Distribution of content priority of different video sequence.

$$P_c(k,i)=\frac{D_t(k,i)-D_t^{min}(k)}{D_t^{max}(k)-D_t^{min}(k)} \quad (9)$$

where $D_t(k,i)$ is the transmission distortion of the ith video packet in the kth video session, $D_t^{max}(k)$ and $D_t^{min}(k)$ are the maximum and minimum transmission distortion of all video packets in the video session, respectively. Note that, in this work, we consider streaming of pre-encoded videos. These information are readily available at the encoder or server end.

Fig. 8 shows the distributions of the content priority defined in (9) for eight test video sequences. Interestingly enough, although these videos have dramatically different scene characteristics, their distributions of content priority (normalized transmission distortion) are very similar, exhibiting an approximately exponential behavior. Tong observed in [8] that, in wavelet- based video encoders, the transmission distortion of video packets is approximately an exponential function of the packet index. It should be noted that these two observations are different. The former is about the probability distribution of the packet-level transmission distortion for hybrid video encoder, while the latter is about the packet-level transmission distortion as a function of the packet index for wavelet-based video encoders.

B. Network Transmission Priority

During video transmission, packets need to arrive at the decoder before their scheduled delay deadlines. Otherwise, they will be considered as lost. Some recent works [3], [4], [28] suggest the following guidelines for defining network transmission priority. *First*, those video packets which are lagging behind their schedule should be assigned with higher network transmission priority levels. *Second*, it is beneficial to drop those packets which are far behind their transmission schedules or most likely to miss their delay deadlines in advance. This can reduce congestion or transmission delays in subsequent links [2].

To this end, we consider the network conditions of subsequent links and estimate the delay each packet might suffer from current node to its destination node. As in [19], [20], and [33], the expected delay can be provided by link-layer network feedback and relayed back hop by hop along its path from destination node to the current node. To reduce the communication overhead, we choose to feedback the average queuing delay of previously delivered packets (within a sliding time window) at the node:

$$\overline{QD}_k^n = \sum_{i=1}^{n_{k0}} QD_k^{n,i} \tag{10}$$

where $QD_k^{n,i}$ denotes the delay of the ith previous packet of Session k experienced at Node n. $n_{k0} = \min(n_k, n_w)$ with n_k being the number of packets in the buffer of current node and n_w the size of the sliding window. In our experiments, we set its value to be 10.

Based on this information, the expected delay for the packet of Session k from current node n to its destination node ED_k^n can be estimated as sum of the delay at each intermediate node, with the assumption the transmission time of each packet is negligible when compared to the queuing delay:

$$ED_k^n = \sum_{i=n}^{N_k} \overline{QD}_k^i \tag{11}$$

or it can be computed recursively as

$$ED_k^n = ED_k^{n+1} + \overline{QD}_k^n \tag{12}$$

where N_k is the total number of nodes in session k, i.e., Node N_k is the destination of Session k. We recognize that, this average queuing delay of different packets with different priorities, used to estimate the expected delay for each packet, might not be accurate due to errors in transmission distortion estimation. Thus, we introduce a relaxation parameter γ, attempting to reduce the sensitivity of packet scheduling to transmission distortion errors.

Based on this estimated delay information, similar to the ideas in [33], the video packets in the buffer of each node can be classified to three groups: 1) *Group A*, packets which have already missed their delay deadlines, i.e., $t - BT_{k,i} \geq D_k$, with t being current time instance and $BT_{k,i}$ the birth time of the ith video packet in Session k. Definitely, they need to be dropped, and this can be done by setting its network transmission priority as 0 as in the first case in (13). 2) *Group B*, packets that have not yet exceeded their delay deadlines and have high probabilities in arriving at the decoder before the delay deadlines $BT_{k,i} + ED_k^n \leq D_k$, or with an overhead less than the given relaxation parameter γ: $BT_{k,i} + ED_k^n \leq D_k + \gamma$. This means that the delay deadline might be satisfied by speeding up its transmission by assigning a higher transmission priority. These two types of packets are most likely to arrive at their destination before their delay deadlines with a proper scheduling algorithm, with a priority inversely proportional to the remaining time. This is the second case in (13). Note, without loss of generality, here the inverse proportion function is used to compute the transmission priority as a function of the remaining time. Any other decreasing functions can also be used. 3) *Group C*, packets that have not yet exceeded their delay deadlines but are most likely to miss their delay deadlines with an overhead larger than the relaxation parameter. We will drop them in advance to reduce congestion in subsequent links. This is the third case in (13).

Based on these observations, the network transmission priority for the ith video packet in video session k at node n at time t can be defined as (13) at the bottom of the page, where t is the current time, D and BT are the delay deadline and birth time of the packet, respectively.

C. Packet Scheduling for Multi-Session Video Streaming Over Multi-Hope Mesh Networks

In order to form a compound metric to measure the overall packet scheduling priority, we propose to use the following formula:

$$P(k,i,n,t) = [P_c(k,i)]^{\mu} \times [P_s(k,i,n,i)]^{v} \tag{14}$$

where $P_c(k,i)$ and $P_s(k,i,n,t)$ are the content priority and network transmission priority defined in (9) and (13). μ and v are two algorithm control parameters. For example, if one of them is set to 1 and the other to 0, the algorithm will degrade to pure content-aware scheduling, similar to the highest value first scheme [29] or deadline- aware scheduling scheme, i.e., early deadline first (EDF) [28]. When both parameters are 0's, neither the content diversity nor the deadline constraints is considered and it becomes the original FIFO (first-in-first-out, or FCFS, first come first serve) scheduler. The impact of parameters μ *and* v on the overall video streaming performance will be evaluated in Section V.

We assume that an overlay network topology [20] is deployed to provide feedback information about the average queuing delay at each node along the transmission path of each video session, which is needed in (10)–(13) for computing the network transmission priority. The content priority $P_c(k,i)$ is encoded as a part of the packet header information. The network transmission priority is computed at each node with current feedback information about the average queuing delays of subsequent links. According to (14), the overall scheduling

$$P_s(k,i,n,t) = \begin{cases} 0, & \text{if } t - BT_{k,i} \geq D_k \\ \frac{1}{D_k - \left[(t-BT_{k,i}) + ED_k^n\right] + \gamma_k^n + 1} & \text{if } \left[(t-BT_{k,i}) + ED_k^n\right] - D_k \leq \gamma_k^n \\ 0, & \text{otherwise} \end{cases} \tag{13}$$

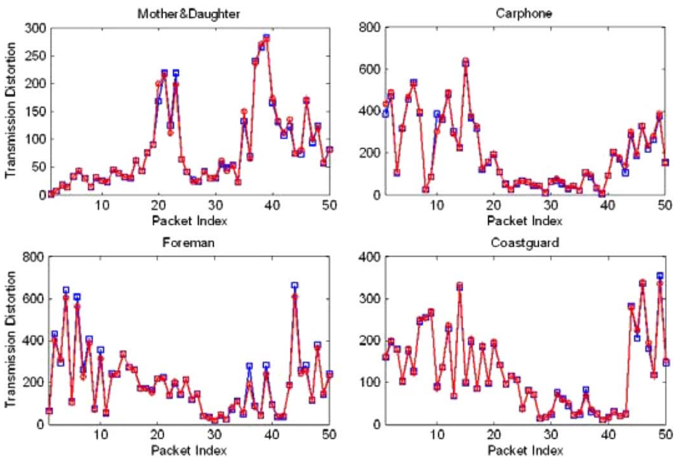

Fig. 9. Estimation of the transmission distortion for each packet of four test videos.

priority can be computed for each packet in its transmission queue at each node and the packet with the highest scheduling priority is then selected to be served in the next transmission interval. In case of buffer overflow at intermediate nodes due to heavy traffic load or low bandwidth, those packets with the least scheduling priorities are chosen to be dropped.

TABLE IV
RELATIVE ESTIMATION ERROR IN TRANSMISSION DISTORTION

Video	REE	Video	REE
Mother&Daughter	1.80%	Salesman	0.92%
Carphone	2.48%	Akiyo	1.12%
Foreman	4.77%	News	1.26%
Coastguard	1.84%	Table tennis	6.08%
Miss America	0.33%	Highway	1.80%
Silent	3.05%	Hall Monitor	2.52%

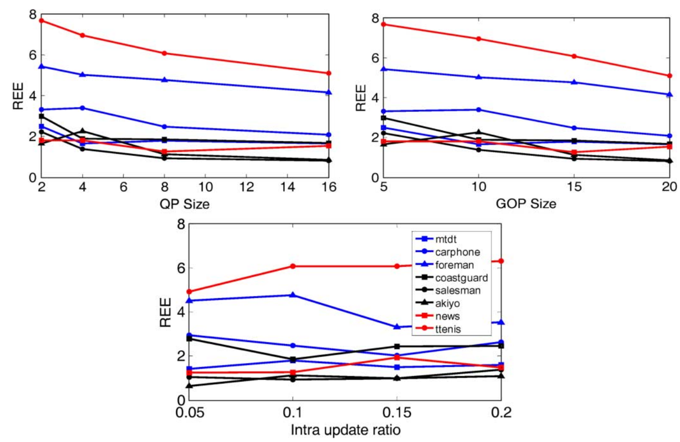

Fig. 10. REE of transmission distortion at different encoder settings.

V. PERFORMANCE EVALUATION

In this section, we evaluate the performance of the proposed packet-level transmission distortion model and the proposed content-and-deadline-aware packet scheduling scheme.

A. Performance Evaluation of the Packet-Level Transmission Distortion Model

We implement the proposed transmission distortion estimation algorithm in an MPEG-4 video encoder [11]. (In our future work, we plan to implement it in an H.264 video encoder.) The following 12 CIF videos: Mother & Daughter, Carphone, Foreman, Coastguard, Salesman, Akiyo, News and Tabletennis, Miss-America, Highway, Silent, and Hall-Monitor, are used for performance evaluation. The packet size is set to be 500 bytes (aligned to the MB boundary).

1) Packet-Level Transmission Distortion Estimation Results: For each video packet, we use the algorithm presented in Section III to estimate its transmission distortion. To obtain the ground truth, we compute the distortion at the decoder side when this packet is dropped. Fig. 9 shows the estimated transmission distortion (in circles) in comparison with its ground truth (in squares) for each packet of four test videos. Due to the limited space, we only show the results for the first 50 packets. Simulations over other video packets yield similar results. To numerically measure the estimation accuracy, we define the relative estimation error (REE) as follows:

$$REE = \frac{|D_t - \widehat{D}_t|}{D_t} \times 100\% \qquad (15)$$

where D_t and $\widehat{D}_t$ are actual and estimated transmission distortion, respectively.

Table IV shows the REE for all frames of all 12 test video sequence. It can be seen that the proposed packet-level transmission distortion is very accurate and the REE is mostly less than 3%.

2) Transmission Distortion Estimation Under Different Encoder Settings: Note that the amount of transmission distortion depends on the encoder settings, such as QP and GOP size, and intra refresh ratio. This is because the QP determines the quality of reconstructed video frames, and the GOP size and intra refresh ration control the amount of error propagation. In the following experiments, we evaluate the performance of the proposed transmission distortion model under different encoder settings. Fig. 10 shows the relative estimation error of transmission distortion at different settings of QP, GOP size and intra refresh ratio for the first eight test video sequences. We can see that the proposed algorithm achieves fairly accurate and robust estimation at different encoder settings. As GOP size increases, the transmission error might propagate to more frames, which yields a large transmission distortion. However, the estimation error caused by our model has not increased much. Therefore, REE, which is the ratio between them, becomes smaller. This also holds for increasing QPs. However, for the case of Intra refreshing ratio, we observe that the pattern is not clear.

3) Transmission Distortion Estimation With Different Error Concealment Schemes: In the above discussion, we chose the simple copy-from- previous-frame error concealment scheme as in [16], [22], [25], and [32]. However, we claim that the proposed model is applicable with different error concealment schemes. As shown in (8), there are two variables, namely the instantaneous transmission distortion (ITD) and the motion reference ratio (MRR). The ITD represents the distortion introduced

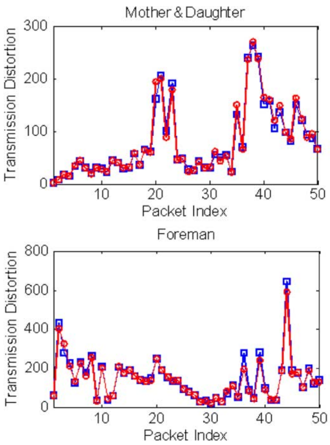

Fig. 11. Estimation results with complex error concealment.

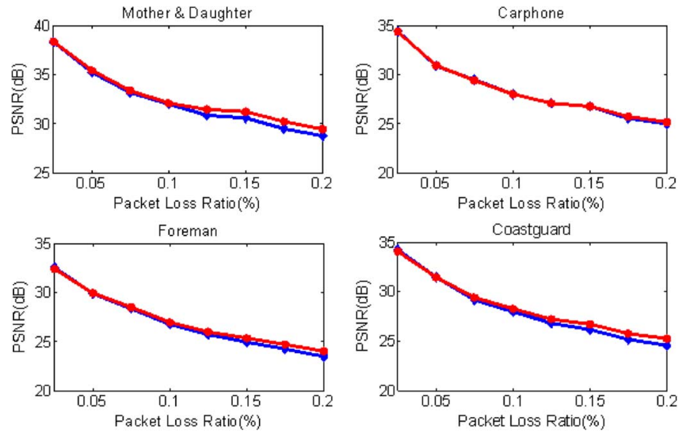

Fig. 12. Verification of additivity for packet-level transmission distortion.

in the current frame due to error concealment. The MRR characterizes the error propagation behavior. We observe that only the ITD depends on the specific error concealment scheme while the MRR only depends on the scene characteristics of the video. To demonstrate the performance of our transmission distortion estimation on different error concealment schemes, we have implemented the error concealment scheme of motion compensated temporal prediction [32]. The results are shown in Fig. 11. We can see that the estimated results match the actual values very well. This again shows the effectiveness and robustness of the proposed transmission distortion model.

4) Uncorrelated Transmission Errors Between Different Video Packets: In this section, we verify the assumption that transmission errors within different video packets are uncorrelated with each other. To this end, we simulate different scenarios of packet loss with packet loss ratios up to 20%. For each scenario, packets are randomly chosen to be lost. The video is reconstructed and the corresponding transmission distortion (in terms of PSNR) is calculated at the decoder. This serves as the ground-truth for transmission distortion estimation. For comparison, we estimate the transmission distortion of each lost packet and use their sum as the estimated transmission distortion. Each experiment is repeated for 50 times and the average PSNR value is reported.

The experimental results of four test video sequences are shown in Fig. 12. We can see that the estimation is very close to the ground-truth values at different packet loss ratios. Experiments over other videos yield similar results. Therefore, we conclude that transmission errors within different video packets are uncorrelated with each other and we can estimate the transmission distortion with different packet losses using the sum of packet-level transmission distortion [25].

B. *Performance Evaluation of the Proposed Scheduling Policy*

We use a discrete event simulator to simulate multi-session video transmission over a multi-hop mesh network and implement the CDAS scheme developed in Section IV at each node. We assume an overlay network topology [20] is deployed to collect feedback information about the average queuing delay at subsequent nodes and provide these statistics to each node [33]. To consider the interference among nodes during video streaming over wireless mesh networks, we use a K-hop interference model [30], [31], in which any two nodes within K-hop distance cannot transmit simultaneously. This can be characterized by an interference matrix C, which indicates the set of nodes that can transmit data simultaneously without interference [30]. In our simulations, we set K to be one hop. For transmission scheduling and link-layer control, the widely used greedy maximal scheduling (GMS) algorithm [30], [31] is employed. In GMS, the scheduler is determined by choosing links in a decreasing order of the backlog while conforming to interference constraints. More specifically, at each time slot, the node with the largest backlog is first chosen to be active, and all nodes interferencing with the chosen link are deactivated. Then the node with the largest backlog among the remaining nodes is selected and this process continues until no node left. Then, the proposed CDAS algorithm is carried out for those active nodes.

Fig. 13 shows a random network topology with 50 nodes generated with our topology generator. We assume that the mesh network topology is fixed over the duration of the video session. We simulate five video sessions simultaneously being transmitted over the network. These five test video sequences, Mother & Daughter, Carphone, Foreman, Coastguard, and Salesman, all in QCIF format, are encoded by an MPEG-4 encoder [11] at 30 fps. During the encoding process, we collect the motion reference ration information and estimate the transmission distortion and content priority for each packet, and encode the content priority information in the packet header.

For each video session, we randomly select its source and destination node and use a shortest-path routing scheme to determine the route for each session, as shown in Fig. 13, with each colored line representing the path of a video session while the red circles and blue squares denoting the source and destination nodes respectively. Note that a number of nodes and links are shared by two or multiple video sessions. The proposed packet scheduling scheme operates at each node, determining which packet to be served during the next transmission interval. The received video bit streams are decoded at the receiver side. For those video packets dropped by the packet scheduler or those received behind the scheduled playback time, the decoder considers them as lost and conducts error concealment. We then

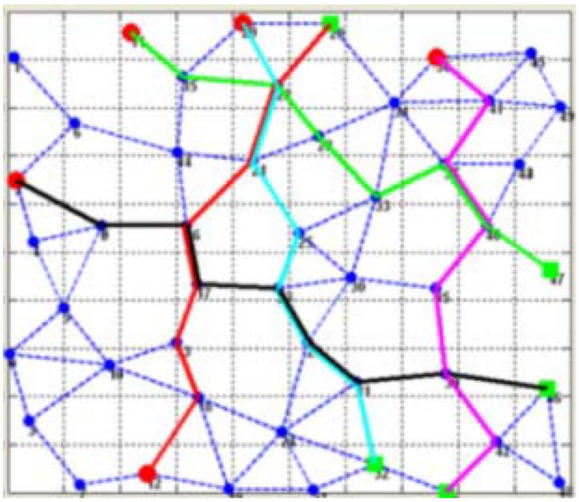

Fig. 13. Randomly generated mesh network topology with five video sessions.

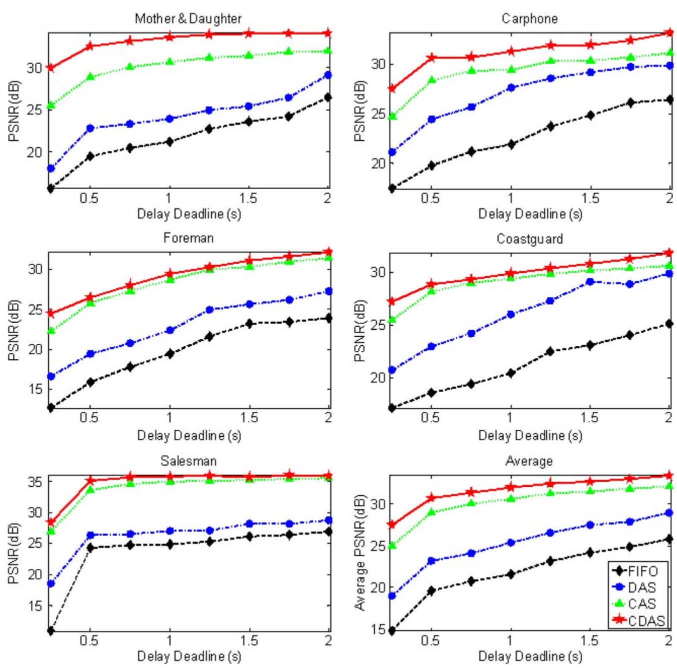

Fig. 14. Comparison of the reconstructed video quality.

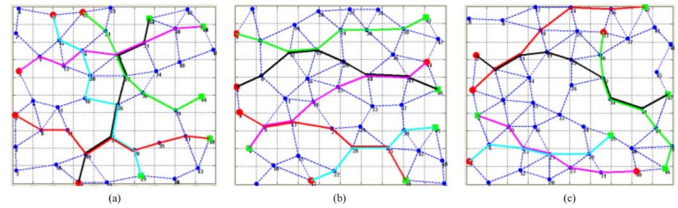

Fig. 15. Different network topologies. (a) Topology2. (b) Topology3. (c) Topology4.

TABLE V
AVERAGE PSNR (dB) OF THE RECONSTRUCTED VIDEOS WITH TOPOLOGY 2

Delay Deadlines(s)	FIFO	DAS	CAS	CDAS
0.25	15.5273	20.3829	27.3531	**29.5407**
0.50	20.9809	24.7542	31.3152	**32.4200**
0.75	22.1713	26.3984	32.4819	**33.4312**
1.00	24.6100	28.6059	33.0410	**33.7963**
1.25	26.0729	30.1999	33.5157	**34.1936**
1.50	26.8658	31.3167	33.7646	**34.3996**
1.75	27.8604	32.3663	34.0203	**34.7724**
2.00	30.4882	34.1635	34.2415	**35.0400**

TABLE VI
AVERAGE PSNR (dB) OF THE RECONSTRUCTED VIDEOS WITH TOPOLOGY 3

Delay Deadline(s)	FIFO	DAS	CAS	CDAS
0.25	18.7107	23.6299	28.3905	**30.0243**
0.50	20.5061	25.7009	30.9499	**31.8077**
0.75	22.0267	27.1233	31.9663	**32.6033**
1.00	24.0865	28.8659	32.5560	**33.1422**
1.25	25.7550	30.1721	33.0487	**33.5137**
1.50	26.5110	31.2071	33.3774	**33.8932**
1.75	27.0797	32.3431	33.7040	**34.2735**
2.00	30.3910	33.6058	33.9096	**34.6385**

TABLE VII
AVERAGE PSNR (dB) OF THE RECONSTRUCTED VIDEOS WITH TOPOLOGY 4

Delay Deadline(s)	FIFO	DAS	CAS	CDAS
0.25	18.4315	23.8814	29.2371	**30.8989**
0.50	21.0829	26.4862	31.3464	**32.5193**
0.75	22.1465	28.2101	32.3173	**33.4165**
1.00	24.7816	30.2060	32.8861	**33.7446**
1.25	26.4221	31.6150	33.4128	**34.1647**
1.50	26.8462	32.2491	33.6494	**34.4324**
1.75	27.8812	33.1779	33.9482	**34.7545**
2.00s	31.1327	34.1481	34.1845	**35.0092**

measure the MSE between the decoder reconstruction and the original video and compute the average PSNR.

We compare our proposed scheduling scheme with the following methods: 1) FIFO which has been widely used in many existing applications, the same as the sequential scheduling (SS) in [9]; 2) delay-aware scheduling (DAS) [with $\mu = 0$ and v $= 1$ in (11)]; and 3) content-aware scheduling (CAS) which only considers the content priority [with $\mu = 1$ and v $= 0$ in (13)].

1) Experimental Results of Proposed CADS Scheme: Fig. 14 shows the average PSNR results of the four packet scheduling schemes for all five test videos at different delay deadlines. We can see that the proposed packet scheduling scheme consistently outperforms other methods. We also show the average PSNR of all five test videos in the last plot of Fig. 14.

Fig. 15 shows three additional randomly generated network topologies with different sender-receiver configurations. The simulations results are summarized in Tables V–VII. We can see that for different network topologies, an average of 3–6 dB gain for small to moderate delay deadline and 1 dB gain for large delay deadline in video quality is achieved.

2) Sensitivity Analysis of Parameters μ and v: In the following experiment, we evaluate the impact of control parameters μ and v on the performance of the proposed scheme on Network Topology 1. Table VIII shows the average PSNR of all test videos at different delay deadline with $v = 1$ and μ changing from 0.1 to 10. Table IX shows the average PSNR of all test videos at different delay deadline with $\mu = 1$ and v changing from 0.1 to 10. As can be observed, the scheduling policy with $\mu = 1$ and $v = 1$ yields the best performance in most cases.

3) Impact of the Transmission Distortion Estimation Error: The proposed packet scheduling scheme relies on transmission

TABLE VIII
AVERAGE PSNR (dB) OF THE RECONSTRUCTED VIDEOS WITH $\upsilon = 1$

Delay Deadline(s) \ μ	0	0.1	1	10
0.25	19.0015	26.6166	27.5166	**28.0141**
0.50	23.2033	29.5870	**30.7030**	30.6497
0.75	24.0941	30.0726	**31.3690**	31.3610
1.00	25.3867	30.6904	**32.0088**	31.7749
1.25	26.5787	31.3016	**32.4872**	32.2750
1.50	27.5144	31.8205	**32.7282**	32.4681
1.75	27.8899	32.4874	**33.0631**	32.8135
2.00	28.9805	32.9750	**33.4363**	33.0062

TABLE IX
AVERAGE PSNR (dB) OF THE RECONSTRUCTED VIDEOS WITH $\boldsymbol{\mu} = 1$

Delay Deadlines(s) \ υ	0	0.1	1	10
0.25	27.9541	**28.0141**	27.5166	26.6166
0.50	30.6310	30.6497	**30.7030**	29.5870
0.75	31.3687	31.3610	**31.3690**	30.0726
1.00	31.7552	31.7749	**32.0088**	30.6904
1.25	32.2527	32.2750	**32.4872**	31.3016
1.50	32.3928	32.4681	**32.7282**	31.8205
1.75	32.7763	32.8135	**33.0631**	32.4874
2.00	32.9696	33.0062	**33.4363**	32.9750

TABLE X
IMPACT OF THE TRANSMISSION DISTORTION ESTIMATION ERROR

Delay Deadlines(s) \ REE	0	3%	6%	9%	12%	15%
0.25	27.5166	**28.0955**	27.4200	26.6694	26.4899	26.6039
0.50	**30.7030**	30.1812	29.5016	28.9063	28.5532	28.2704
0.75	**31.3690**	30.7149	29.9681	29.5917	29.1562	28.9501
1.00	**32.0088**	31.2584	30.5361	29.9754	29.6760	29.3070
1.25	**32.4872**	31.6185	30.9108	30.3175	30.0227	29.6339
1.50	**32.7282**	31.9543	31.2242	30.5821	30.2454	29.7718
1.75	**33.0631**	32.1562	31.4395	30.9112	30.4987	30.0064
2.00	**33.4363**	32.6137	31.7022	31.0877	30.7830	30.2813

distortion modeling and prediction at the encoder side. Certainly, its performance depends on the accuracy in transmission distortion estimation. In the following experiments, we study the impact of the transmission distortion estimation error on the performance of the proposed scheduling scheme. To this end, we introduce a random noise to the ground-truth transmission distortion to increase the relative estimation error. The average video quality of test videos on Topology 1 for different delay deadlines with different levels of REEs is summarized in Table X. As REE increases from 0% to 15%, the maximum PSNR degradation is about 3 dB. From Section V-A, we can see that our transmission distortion estimation achieves an REE of less than 3% for most videos, which leads to a quality loss less than 0.8 dB. This gives us a satisfied level of confidence in our proposed transmission distortion estimation and packet scheduling scheme.

VI. CONCLUSION AND FURTHER DISCUSSION

In this work, we have developed a fine-granularity transmission distortion model for the encoder to predict the quality degradation of decoded videos caused by lost video packets. Based on this packet-level transmission distortion model, we have established a content-and-deadline-aware scheduling scheme for multi-session video streaming over multi-hop mesh networks, where content priority, queuing delays, and dynamic network transmission conditions of video packets are jointly considered. Our extensive experimental results demonstrate that the proposed distortion model and the CDAS scheme significantly improve the performance of multi-session video streaming over mesh networks.

In our future work, based on the packet-level transmission distortion estimation and scheduling algorithms, we will jointly study admission/congestion control with packet scheduling, and further extend it to a cross-layer resource allocation and performance optimization framework for video streaming over mesh networks. We also plan to extend the transmission distortion model to H.264 video coding and evaluate the packet scheduling and cross-layer performance optimization.

REFERENCES

[1] P. V. Pahalawatta and A. K. Katsaggelos, "Review of content-aware resource allocation schemes for video streaming over wireless networks," *Wireless Comm. Mob. Comput.*, vol. 7, no. 2, pp. 131–142, Feb. 2007.

[2] H. Shiang and M. van der Schaar, "Multi-user video streaming over multi-hop wireless network: A distributed, cross-layer approach based on priority queuing," *IEEE J. Select. Areas. Commun.*, vol. 25, no. 4, pp. 770–786, May 2007.

[3] A. Dua and N. Bambos, "Downlink wireless packet scheduling with deadlines," *IEEE Trans. Mob. Comput.*, vol. 6, no. 12, pp. 1410–1425, Dec. 2007.

[4] Y. Ofuji, S. Abeta, and M. Sawahashi, "Unified packet scheduling method considering delay requirement in forward link broadband wireless access," in *Proc. IEEE 58th Vehicular Technology Conf.*, Oct. 2003.

[5] R. Agrawal, V. Subramanian, and R. Berry, "Joint scheduling and resource allocation in CDMA systems," *IEEE Trans. Inf. Theory*, to be published.

[6] J.-R. Ohm, "Advances in scalable video coding," *Proc. IEEE*, vol. 93, no. 1, pp. 42–56, Jan. 2005.

[7] M. van der Schaar, Y. Andreopoulos, and Z. Hu, "Optimized scalable video streaming over IEEE 802.11a/e HCCA wireless networks under delay constraints," *IEEE Trans. Mob. Comput.*, vol. 5, no. 6, pp. 755–768, Jun. 2006.

[8] X. Tong, Y. Andreopoulos, and M. van der Schaar, "Distortion-driven video streaming over multihop wireless networks with path diversity," *IEEE Trans. Mob. Comput.*, vol. 6, no. 12, pp. 1343–1356, Dec. 2007.

[9] Z. Miao and A. Ortega, "Expected run-time distortion based scheduling for delivery of scalable media," in *Proc. Packet Video*, Apr. 2002.

[10] P. A. Chou and Z. Miao, "Rate-distortion optimized sender driven streaming over best-effort networks," in *Proc. IEEE Workshop Multimedia Signal Processing*, Oct. 2001, pp. 587–592.

[11] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 7, no. 1, pp. 19–31, Feb. 1997.

[12] R. Tupelly, J. Zhang, and E. Chong, "Opportunistic scheduling for streaming video in wireless networks," in *Proc. Conf. Information Sciences and Systems*, 2003.

[13] S. Kang and A. Zakhor, "Packet scheduling algorithm for wireless video streaming," in *Proc. Packet Video*, Apr. 2002.

[14] P. Pahalawatta, R. Berry, T. Pappas, and A. Katsaggelos, "Content-aware resource allocation and packet scheduling for video transmission over wireless networks," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 749–759, May 2007.

[15] I. Politis, T. Pliakas, and M. Tsagkaropoulos *et al.*, "Distortion optimized scheduling and QoS driven prioritization of video streams over WLAN," in *Proc. Packet Video*, 2007, pp. 349–355.

[16] J. Chakareski and P. Frossard, "Rate-distortion optimized distributed packet scheduling of multiple video streams over shared communication resources," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 207–218, Apr. 2006.

[17] H. Zhang and S. Rangarajan, "Adaptive scheduling of streaming video over wireless networks," in *2008 IEEE Int. Conf. Multimedia and Expo.*, Apr. 2008, pp. 477–480.

[18] E. Setton and B. Girod, "Congestion-distortion optimized scheduling of video over a bottleneck link," in *Proc. IEEE Int. Workshop on Multimedia Signal Processing*, Sep. 2004.

[19] M. Kodialam and T. Nandagopal, "Characterizing achievable rates in multi-hop wireless networks: The joint routing and scheduling problem," in *Proc. ACM Int. Conf. Mobile Computing and Networking*, 2003, pp. 42–54.

[20] L. Xiao, M. Johansson, and S. P. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1136–1144, Jul. 2004.

[21] F. Zhang, T. C. Todd, D. Zhao, and V. Kezys, "Power saving access points for IEEE 802-11 wireless network infrastructure," *IEEE Trans. Mobile Comput.*, vol. 5, no. 2, pp. 144–156, Feb. 2006.

[22] Z. He and H. Xiong, "Transmission distortion analysis for real-time video encoding and streaming over wireless network," *IEEE Trans. Circuit. Syst. Video. Technol.*, vol. 16, no. 9, pp. 1051–1062, Sep. 2006.

[23] C. Zhang, H. Yang, S. Yu, X. Yang, and H. Liu, "GOP-level transmission distortion modeling for unequal importance judgment," in *2007 Proc. IEEE Int. Conf. Multimedia and Expo.*, Jul. 2007, pp. 855–858.

[24] S. Nyamweno, R. Satyan, and F. Labeau, "Exponential decay of transmission distortion in H.264," in *Proc. IEEE 9th Workshop Multimedia Signal Processing*, 2007, pp. 268–271.

[25] K. Stuhlmuller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Select. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.

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

[27] H. Chiou, Y. Lee, and C. Lin, "Content-aware error-resilient transcoding using prioritized intra-refresh for video streaming," *J. Vis. Comm. Image Represent.*, vol. 16, no. 4–5, pp. 563–588, Apr. 2005.

[28] E. D. Jensen, C. D. Locke, and H. Toduda, "A time-driven scheduling model for real-time operating systems," in *Proc. 6th IEEE Real-Time Systems Symp.*, Dec. 1985, pp. 112–122.

[29] C. L. Liu and J. W. Layland, "Scheduling algorithms for multiprogramming in a hard real-time environment," *J. Assoc. Comput. Mach.*, vol. 20, no. 1, pp. 46–61, Jan. 1973.

[30] C. Joo, X. Lin, and N. B. Shroff, "Understanding the capacity region of the greedy maximal scheduling algorithm in multi-hop-wireless networks," in *Proc. IEEE Conf. Computer Communications*, Apr. 2008, pp. 1103–1111.

[31] C. Joo, X. Lin, and N. B. Shroff, "Performance limits of greedy maximal matching in multi-hop wireless networks," in *Proc. 46th IEEE Conf. Decision and Control*, Dec. 2007, pp. 1128–1133.

[32] 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.

[33] H.-P. Shiang and M. van der Schaar, "Informationally decentralized video streaming over multihop wireless networks," *IEEE Trans. Multimedia*, vol. 9, no. 6, pp. 1299–1313, Oct. 2007.

Yongfei Zhang received the B.S. degree in electrical engineering from Beihang University, Beijing, China, in 2005. He is currently pursuing the Ph.D. degree in electrical and computer engineering at Beihang University.

He has been a visiting student for two years in the Department of Electrical and Computer Engineering, University of Missouri, Columbia. His current research is focused on image/video processing and communication, packet scheduling, resource allocation, and performance optimization for video transmission over mesh networks. His research interests also include pattern recognition, intelligent control, and optimization.

Shiyin Qin received the B.S. and M.S. degrees in engineering science in automatic controls and industrial systems engineering from Lanzhou Jiaotong University, Lanzhou, China, in 1978 and 1984, respectively, and the Ph.D. degree in industrial control engineering and intelligent automation from Zhejiang University, Hangzhou, China, in 1990.

He has been a Professor at Xi'an Jiaotong University, Xi'an, China, and Beijing University of Technology, Beijing, China. He is now a Professor at the School of Automation Science and Electrical Engineering at Beihang University in Beijing. His current research interests include intelligent autonomous controls of complex spacecrafts; autonomous intelligent controls of formation process for multi-robot system; the theory of hybrid control systems with its applications; fault diagnosis and fault-tolerant controls; image processing and pattern recognition; intelligent systems and artificial life; the modeling and optimizing decision of open complex giant systems; and computational intelligence and entropy optimization. He has published more than 140 papers. He is also the author of three monographs.

Dr. Qin is an outstanding member of the council and the secretary-general of Chinese Association for Artificial Intelligence (CAAI), the vice-chairman of the Society of Intelligent Control and Intelligent Management in CAAI, and is also a member of the Committee of Intelligent Automation of Chinese Association of Automation (CAA). He was awarded the First Level Prize of "1999 National Excellent Books of Science and Technology and the Progress of Science and Technology", and the Gold Medal Prize of the Excellent Software of "the 5th National Engineering Design" (1999).

Zhihai He (S'98–M'01–SM'06) received the B.S. degree from Beijing Normal University, Beijing, China, and the M.S. degree from Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, in 1994 and 1997, respectively, both in mathematics, and the Ph.D. degree in electrical engineering from University of California, Santa Barbara, in 2001.

In 2001, he joined Sarnoff Corporation, Princeton, NJ, as a Member of Technical Staff. In 2003, he joined the Department of Electrical and Computer Engineering, University of Missouri, Columbia, as an Assistant Professor. His current research interests include image/video processing and compression, network transmission, wireless communication, computer vision analysis, sensor networks, and embedded system design.

Dr. He received the 2002 IEEE Transactions on Circuits and Systems for Video Technology Best Paper Award and the SPIE VCIP Young Investigator Award in 2004. Currently, he serves as an Associate Editor for IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY, IEEE TRANSACTIONS ON MULTIMEDIA, and *Journal of Visual Communication and Image Representation*. He is also a guest-editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Special Issue on Video Surveillance. He is a member of the Visual Signal Processing and Communication Technical Committee of the IEEE Circuits and Systems Society, and serves as Technical Program Committee member or session chair of a number of international conferences.