Content-Aware Distortion-Fair Video Streaming in Congested Networks

Ying Li, *Student Member, IEEE*, Zhu Li, *Senior Member, IEEE*, Mung Chiang, *Senior Member, IEEE*, and A. Robert Calderbank, *Fellow, IEEE*

Abstract—**Internet is experiencing a substantial growth of video traffic. Given the limited network bandwidth resources, how to provide Internet users with good video playback quality-of-service (QoS) is a key problem. For video clips competing bandwidth, we propose an approach of Content-Aware distortion-Fair (CAF) video delivery scheme, which is aware of the characteristics of video frames and ensures max-min distortion-fair sharing among video flows. CAF leverages content-awareness to prioritize packet dropping during congestion. Different from bandwidth fair sharing, CAF targets end-to-end video playback quality fairness among users. The proposed CAF approach does not require rate-distortion modeling of the source, which is difficult to estimate. Instead, it exploits the temporal prediction structure of the video sequences along with a frame drop distortion metric to guide resource allocations and coordinations. Experimental results show that the proposed approach operates with limited overhead in computation and communication, and yields better QoS, especially when the network is congested.**

Index Terms—**Fairness, Internet, multimedia communication, optimization, rate distortion, resource allocation, video.**

I. INTRODUCTION

A. Motivation

THE recent advances in transmission, display, and storage capabilities in the Internet and end-user devices have ushered an unprecedented growth of video traffic over the Internet. IPTV, user-generated videos, and video-based social networking are driving up this video traffic demand on the network everyday. Unlike web and e-mail traffic, video is characterized by its large bandwidth requirement and stringent quality-of-service (QoS) parameters, especially delays. This presents a challenge to network engineering different from data and voice dominated traffic [35].

In video traffic, certain packets can be dropped if the network is congested, with controllable decoding quality degradation. Transcoding solutions [37] can offer good flexibilities in rate-distortion shaping and the scalable video coding [27], [29], [36] can offer temporal, spatial, and signal-noise-ratio (SNR) scalability. All these technologies carry out the adaptive video source coding at the application layer, providing the opportunity of smartly dropping frames or packets when the communication links are congested or bandwidth is limited.

For the problem of transporting video traffic over a network, the design principal is to coordinate network layer resource allocation with the application layer video adaptation schemes, such that a graceful trade-off between resource consumed and QoS delivered is achieved. Advances in video signal processing and coding have made a rich set of video coding and adaptation tools available to content delivery network engineering.

Traditionally the content generation and the network resource engineering are designed separately, following a content-pipe divide [9]. Network engineering traditionally is not content-aware, treating all bits the same. For video traffic, different frames may have different importance and can contribute differently in terms of the distortion perceived by human eyes. Dropping some less important frames may only have little influence on the perceptual video distortion, if the important frames could have better guarantee to be transmitted successfully. Content-awareness can be utilized to improve the QoS provision. In this work, we study video streaming over the Internet, and develop a content-aware solution for multihop video communications.

For network engineering, an important issue is congestion control at the transport layer. What is interesting and extensively researched in the last decade is that the problem of Internet congestion control can be cast as an optimization problem with economics interpretation [19], [24], [25], the network utility maximization (NUM), which aims to maximize the aggregate utility of all the users, subject to the constraint that the total flow on each link should be no greater than the link capacity. Here, the utility of each user s with a transmission rate x_s is a strictly concave, continuously differentiable nondecreasing function $U_s(x_s)$, measuring the user's "happiness" when allocated rate x_s. By decomposition method with a pricing interpretation, the optimization problem can be solved via distributed algorithms. Since the publication of the seminal paper [19] in 1998, the framework of NUM has found many applications in network resource allocation algorithms and design of protocol stacks [10]. It has become a mathematical foundation of many cross-layer designs.

Indeed, cross-layer design has shown its advantage for QoS provision and resource allocation. The NUM framework can be extended to video delivery problems, where the utility function

can be the opposite of the distortion function of transmission rate. There are many existing works in cross-layer optimization in video delivery, e.g., [7], [8], [15], [26], [31], [34], and [39]. These works are mainly focusing on the application layer, MAC layer, physical layer, and other layers.

A branch of cross-layer design, content-aware networking, brings the intelligence of video source coding at the application layer and network resource engineering together. A lot of works utilize content-awareness in different scenarios, such as in video summarization [22], packetized video delivery [11], [30], in P2P streaming [3], in wireless networks [16], [21], in content delivery networks [2], and in image authentication [40]. In [3], [11], [16], [22], and [30], the content-awareness lies in the importance of the frames or packets based on the distortion and the dependency of decoding. In [21], content-aware-ness is a metric of the motion characteristics of different scenes. In [2], content-awareness lies in the knowledge of descriptions of the multiple description source coding. In all these works, content-awareness is utilized adaptively fitting in the delivery requirement. For instance, through content analysis and optimization, video summarization schemes such as in [22] and [23] select a subset of frames from the video sequence to form a concise representation of the sequence, to deliver good visual quality at low bit rates. The work for multiaccess wireless video streaming [16] develops a network utility maximization with a resource pricing solution via dual decomposition and the resource price, in turn, drives content adaptation solutions at each user in a distributed fashion. In this work, we focus on the similar content-awareness as in [16] and [22]. Rather than one-hop multiaccess networks in [16], we study in the setting of general multihop Internet, where we have to deal with the dependency of users and links in terms of who are sharing which link.

Another theme of this work is about fairness among competing users. Since motion-rich video clip may require more bandwidth than video clip with less motion, when such users are sharing the resource, bandwidth fairness may not be a good metric. The end users care about the video quality rather than the bandwidth; hence, we focus on distortion-fair allocation, especially max-min fair provision among uses.

B. Related Work

Authors of [5] and [33] assume an explicit utility function of rate, and the algorithms are based on the utility function rather than on content. Studies in [5] and [33] are not specifically for video and they do not take into account the special characteristics of video traffic. For video, the utility function is time-varying and it is hard to estimate the utility function accurately on the fly. In addition, the distributed algorithm in [33] may need a lot of iterations and the rate may fluctuate dramatically and the video quality fluctuates. Different from [5] and [33], we use the utility of every frame, which can be easily and explicitly calculated.

Our work and many studies such as [4], [6], [13], [17], [18], [28], [38], and works mentioned above on cross-layer video delivery and content-aware networking are all in favor of the optimal rate-distortion tradeoff, but our work focuses on the metric of min-max distortion fairness for many users sharing a general multihop network.

Our work has a fine granularity in how to drop frames when the link is congested. In [4], the authors show that the priority dropping frames of layered video can have gain over uniform dropping. The dropping is based on the importance of the layers of the video source coding, while in our work the frame dropping is based on the importance of the frames, not limited to layers.

Our work is mainly for stored video, where the side information of the number of bits and importance of the frames can be calculated in advance. Other works, such as [18], are for real-time video. In [18], the authors propose an adaptive congestion control scheme for real-time packet video transport, where the video rate-distortion needs to be estimated in real time. Our work can be extended to real-time video by allowing some computational complexity to derive the importance and length of frames.

C. Overview

The main contributions of this work are as follows.

1) For multihop multiuser video delivery networks, we propose a framework of content-aware distortion-fair networking with joint video source adaptation and network resource allocation, by combining the intelligence of the temporal scalability of the video source coding at the application layer with the congestion control at the transport layer.

2) The framework makes users drop less important frames when the network is congested. Architecturally we illustrate that the approach of end users dropping less important frames is in general more bandwidth-efficient than the approach of links dropping frames when the network is congested.

3) Based on the framework, we propose an algorithm with a provable and small number of iterations for users and links to cooperate, to achieve the min-max distortion fairness.

4) By simulations using standard video clips, we show the performance improvement achieved by our design.

The rest of this paper is organized as follows. Section II explains the video content adaptation schemes to shape video traffic and to achieve rate and distortion tradeoffs. In Section III we give the CAF video networking formulations and in Section IV we derive a solution. The effectiveness of our solution with simulation results is demonstrated in Section V. Section VI presents conclusions and future work.

II. VIDEO ADAPTATION AND TRAFFIC SHAPING

Video traffic is different from other Internet traffic like data, e-mail, and web. Video traffic is in general elastic in the sense that certain packet losses may incur distortions in playback but not catastrophic. Therefore the utility from end users' point of view is not how many bits are delivered but rather the received sequence playback quality. Advances in video signal processing and coding have made available a rich set of coding [32] and adaptation tools, as well as quality metrics that allow applications to trade off video traffic rate with the playback quality. Transcoding [37] based solution offers the most flexibility and rate efficiency for this task, as it in essence re-encodes the sequence to suit specific limits in traffic profiles. However the cost
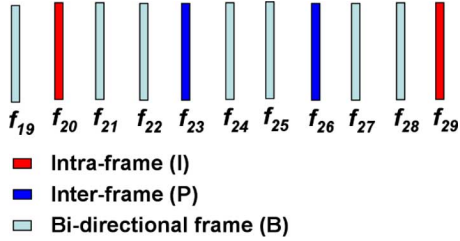
Fig. 1. Group of pictures.

Fig. 2. Distortion: An example for "Foreman" sequence with frame losses.

of transcoding in computation is considerable and it is not suitable for multicast video sessions with large number of receivers. Scalable video coding (SVC) [27] offers the benefit of flexibility in video traffic shaping without transcoding and is well suited for the Internet video delivery. Scalability can be achieved in temporal, frame size, and SNR quality of video.

In this work, we target the temporal scalability of video traffic only, as more advanced scalability features like frame size and SNR scalability are not widely adopted yet, and there exists a large legacy video content coded in MPEG-1/2/4 [1] that has this temporal scalability built in. At the time of congestion over the Internet links, certain frames in the video stream can be dropped with limited loss in playback quality. The goal is how to quantify this loss and have a metric that can allow multiple video sessions to have a quality fair streaming.

Not all video frames are created equal in terms of visual quality impact if lost, as well as the number of bits required to encode. The MPEG [1], [14] coding scheme supports bi-directional prediction frames, or B-frames, which can be dropped without affecting subsequent frame decoding, while if an intra-coded I-frame or predictive coded P-frame is lost, the subsequent frames with prediction on them will not be able to correctly decode.

This is illustrated in Fig. 1, for a group of pictures (GoP) in an MPEG sequence. Consider the GoP with frames 20–28, I-frame f_{20} is independently encoded. Its decoding does not depend on any other frames. Inter-frame f_{23} is decoding dependent upon correct decoding of f_{20}, while f_{26} is dependent on f_{23}. For B-frames, they are decoding dependent on correct decoding of its nearest neighboring I-frames and P-frames, e.g., f_{22}'s decoding depends on f_{20} and f_{23}. Therefore, if an I or P-frame is lost, all frames thereafter in the GoP become un-decodable. So if the system decides to drop an I or P frame, it is equivalent to dropping all frames thereafter to the next GoP start.

To characterize the frame loss distortion, in our previous work [22], [23], we developed a frame drop distortion metric for video rate-distortion video summarization. It measures the distortion to the sequence playback as the difference between the original sequence and the zero-th hold playback of the received frame with frame losses. Given an n-frame sequence $V = \{f_1, f_2, \ldots, f_n\}$, denote the set of frame losses in the sequence as Λ

$$\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_m\}, \quad \lambda_k < \lambda_{k+1} \tag{1}$$

where λ_k is the index of the lost frame. Let the playback sequence be $V = \{f'_1, f'_2, \ldots, f'_n\}$, where playback frame is ei-
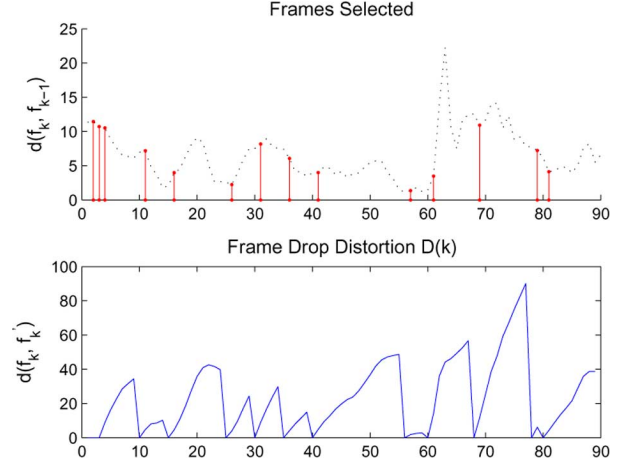
ther the original frame, or the nearest frame that is not lost in the sequence (the latest frame which is not lost prior to the set Λ)

$$f'_k = f_t, \quad t = \arg\max_{j \notin \Lambda} j \leq k \tag{2}$$

then the sequence distortion can be characterized as the distortions incurred by the frame losses during play the playback

$$D(k) = d(f_k, f'_k)$$

where the frame loss distortion metric $d(f_k, f'_k)$ is computed as the principle component analysis (PCA) space distance between scaled frames [22]. An example for "Foreman" sequence with frame losses is plotted in Fig. 2, where the upper sub-figure shows the frame-by-frame distortion $d(f_k, f_{k-1})$, which gives an indication of activity levels in the video sequence, with the stems in the upper sub-figure marking the remaining frames, and the lower sub-figure shows the resulted distortion $D(k)$ due to the frame losses.

Given the set of frame losses as Λ in (1), the resulting video traffic rate reduction because of the frame losses is characterized as

$$R(\Lambda) = \sum_{k=1}^{m} R(f_{\lambda_k}) \tag{3}$$

where $R(f_{\lambda_k})$ is the rate of frame f_{λ_k}. The resulted distortion associated with set Λ can be characterized as

$$D(\Lambda) = \sum_k d(f_k, f'_k) \tag{4}$$

where f'_k is given by (2).

Given the decoding dependency of the frames in the GoP in MPEG sequence, dropping different types of frame can result in different set of frame losses according to the decoding dependency. If the GoP anchor I-frame is lost, then the whole group of pictures is lost. If a P-frame is lost, the frames thereafter in the GoP are useless. If only a B-frame is lost, the rate reduction and the resulted distortion are only limited to the B-frame. For the example in Fig. 1, if I-frame f_{20} is lost, the whole GoP with frames $f_{20} \sim f_{28}$ is lost, if P-frame f_{26} is lost, frames
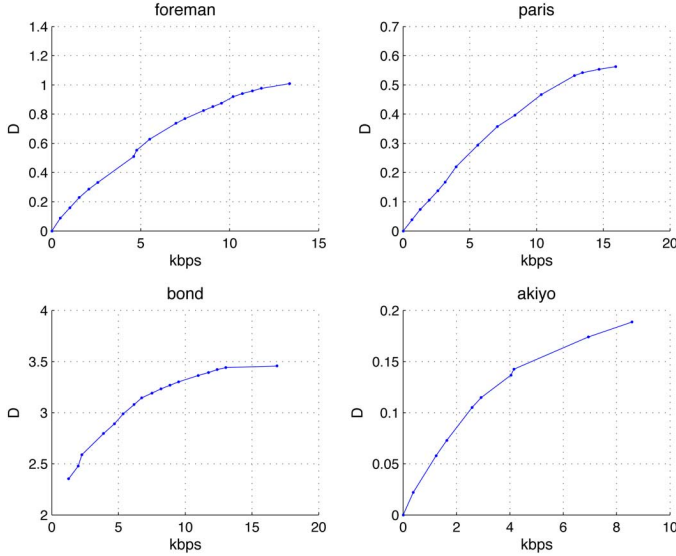
Fig. 3. Rate-saved versus resulted distortion curve. Each iteration gives us a set of frames dropped, the resulted distortion of the drops (the importance or the utility of the dropped frames), and rate reduction.

$f_{26} \sim f_{28}$ should be expunged, and if B-frame f_{27} is lost, only itself is lost. Hence we have the following frame dropping set associated with the frame, taking into account the decoding dependency, for an IBPB GoP structure (for other GoP structure, similar formula can be derived):

$$\Lambda(f_k) = \begin{cases} \{f_k\}, & \text{B frame} \\ \{f_j | j = k-1, \ldots, N_k\}, & \text{P frame} \\ \{f_j | j = k, \ldots, N_k\}, & \text{I frame} \end{cases} \quad (5)$$

where N_k is the last frame of the GOP in which frame f_k lies.

Therefore, for frame f_k, the associated rate reduction is $R(\Lambda(f_k))$ and the resulted distortion is $D(\Lambda(f_k))$.

Obviously, different types and locations of frames in the GoP will have different rate and distortion implications if lost. From information theory, we know that a variety of sources have convex rate-distortion (R-D) functions [12]. This convexity holds true for most practical video source and its operational R-D functions. In practice, there would be a combination of B, P, and I frame drops. Some example of operational R-D curves of the sum distortion of $D(k)$ and loss in rates are plotted in Fig. 3. They indeed exhibit convexity.

Imagine at certain bottleneck link, the video queue needs to be reduced. Armed with the knowledge of R-D operational pair for each frame, we could have a quick sorting of the distortion D, or distortion/rate ratio, D/R, and come up with optimized solutions for either distortion fair (QoS fair), or total distortion minimization. In a network with multiple hops and multiple flows, we develop a distortion fair solution in the following sections.

III. CONTENT-AWARE DISTORTION-FAIR VIDEO STREAMING

A. Max-Min Utility Fair Share

Consider a communication network with L logical links, each with a fixed capacity of c_l bps, and S source-destination pairs, each transmitting at a rate of x_s bps. Each source s emits one flow, using a fixed set $L(s)$ of links, and has a utility function

$U_s(x_s)$. Each link l is shared by a set $S(l)$ of sources. Since the publication of the seminal paper [19] in 1998, the framework of network utility maximization (NUM) has found many applications in network resource allocation algorithms and the design of protocol stacks [10].

NUM, in its basic version, is the problem of maximizing the total utility over the source rates $\mathbf{x}$, subject to linear flow constraints $\sum_{s \in S(l)} x_s \leq c_l$, for every link l. The dual decomposition solutions of the basic NUM [19] give us a distributed solution that iterates on source rate control and link adaptation. The objective of the basic NUM is to achieve social efficiency in utility, but efficiency may not mean fairness, which is very important in resource allocation.

In this work, we look at a different problem: to achieve distortion fairness among video flows, subject to link capacity constraints. Can we have a similar distributed, iterative solution?

The problem we address in this work, utility max-min fair sharing, is the following problem of maximizing the minimum utility, over the source rates $\mathbf{x}$, subject to linear flow constraints for all links:

$$\begin{aligned} \text{maximize} \quad & \min U_s(x_s) \\ \text{subject to} \quad & \sum_{s \in S(l)} x_s \leq c_l, \ \forall l \\ & \mathbf{x} \succeq 0. \end{aligned} \quad (6)$$

B. Utility and Distortion

We focus on the problem of stored video streaming and exploit the IBPB GoP structures in the MPEG coded video. We base our video adaptation and traffic shaping on frame drop mechanisms and distortion metrics discussed in Section II.

For a video session, the video adaptation and traffic shaping are achieved through frame pruning, dropping frames with frame indices $\Lambda = \{\lambda_1, \lambda_2, \ldots, \lambda_m\}$ when there is congestion. The resulting video traffic rate reduction is characterized in (3) and the distortion is characterized in (4).

For a given video segment with n frames, there are total $\sum_{k=1}^{n} \binom{n}{k}$ different combinations of frame drop options. Because of the MPEG coding structure, some frames have dependency in decoding. If an I or P-frame is lost, then the frames from P frame to the end of the GoP are not decodable and therefore shall be pruned also. This is given by (5).

We define the importance of a frame f_k or the utility of a frame f_k, as the resulting distortion summation if this frame is lost, taking into account the decoding dependency

$$D(\Lambda(f_k))$$

where function D is given by (4) and $\Lambda(f_k)$ given by (5).

Consider a typical 15-frame GoP structure $I_1 B_2 P_3 B_4 P_5 B_6 P_7 \ldots B_{14} P_{15}$. If frame P_7 is lost, the frame drops in effect are $\Lambda(P_7) = \{B_6, P_7, B_8, P_9, \ldots, B_{14}, P_{15}\}$, while if a B frame is lost, the resulting pruning is limited to that frame only, i.e., $\Lambda(B_k) = \{B_k\}$.

In this work, instead of optimization over all combinations of frame drops, we only consider the following drop sequences for a given GoP to approximate its rate-distortion curve. Sort all individual frames by its importance, or utility, as the resulting

distortion sum if lost. Then at the time of link adaptation, we start with an empty frame drop set Λ, and keep adding frames to Λ in the increasing order of its distortion $D(\Lambda(f_j))$. This iterative process of traffic shaping is given as

$$\Lambda^0 = \varnothing$$
$$\Lambda^i = \Lambda^{i-1} \cup \{\Lambda(f_{j=\arg\min_{f_j \notin \lambda^{i-1}} D(\Lambda(f_j))})\}. \quad (7)$$

This way, we can approximate R-D curve represented by this "distortion incurred and bits saved" curve, plotted as examples in Fig. 3. For the "Foreman", "Paris", "Bond", and "Akiyo" sequences, we look at a window of 2 s and operate the frame drop adaptation according to (7). For a bit rate reduction in range of 0 to 20 kbps, the resulting distortion incurred is plotted.

Note that in Fig. 3, the resulted distortion actually reflects the importance or the utility of the dropped frames. The shape of the curves in Fig. 3 is the reverse of the traditional R-D curves, because we have the rate as the transmitted rate in the traditional R-D curve, while in Fig. 3, we have the rate reduction as the horizontal axis. The distortion in traditional R-D curve in some sense can be regarded as the opposite of the resulted distortion or the utility/importance of the dropped frames.

C. Min-Max Distortion Fair Share

For aggregate NUM [19], the dual decomposition gives it a distributed solution that allows source to adapt its rate based on congestion prices at links, in an iterative fashion. In this work, we look at the distortion fair formulation, i.e., to give everyone the same QoS as possible, given the network resource constraint. This is given as (8) at the bottom of the page, where Λ_s^l is the frame drops at link l for source s, $L(s)$ is the set of the links that source s uses, $S(l)$ is the set of users who use link l, and R_s^0 is the original data rate for source s without frame drops. In this problem, the first constraint states the flow should not exceed the link capacity, and the second constraint states that for each flow, the set of the frames dropped in the next hop should contain such set at the current hop. The variables are the sets of the frames dropped for each flow at each link it uses.

Instead of optimization over all combinations of frame drops, we only consider the drop sequences as described in (7) for a given GoP to approximate its rate-distortion curve. The min-max distortion fair sharing problem (8) can be simplified with the way we do traffic shaping as in (7) and the curve of the utility of the dropped frames versus reduced rate for video sequences, such as Fig. 3. Denote the resulted distortion

in Fig. 3 as z_s for video flow s. The min-max distortion fair sharing can be stated as to maximize the minimum importance level of z_s among all flows, where the frames with importance less than z_s are pruned for each flow s, and the remaining total flow after pruning of each flow over link l should be no greater than the capacity.

Hence, the problem becomes to find out the threshold z_s of dropping frames for each flow s, where the frames with less importance than the threshold are pruned, such that the link capacities are not exceeded after such pruning. This gives us the solution that is based on the thresholding of the distortion levels.

Note that the traffic shaping as in (7) is based on the importance of frames. Although we draw the R-D curves in Fig. 3, for congestion control, we do not need to send links such curves to let them decide how to drop frames. Instead, the links only need to know the importance of the frames and the length of the frames, because the information of the frame importance and length contains the R-D curve. So for link adaptation, we only need to signal the links with the side information of the importance and number of bits of the frames.

D. Inter-Session Complementary Sharing

Users care about utility, as a function of distortion, rather than bandwidth in a direct way, but bandwidth fair sharing may not always give us utility fairness because the same utility may need different bandwidth according to different content.

Consider two users, each sending a video clip over a common link. When they compete for the bandwidth, by the bandwidth fair sharing, each user gets half of the link capacity. But they may not have the equal satisfaction. For example, say, User 1 is transmitting a video with a lot of motion, like commercial, and User 2 is transmitting a video with less motion, like a person sitting there reading news. The rate-distortion is illustrated in Fig. 4. If the link is congested and some frames of the video should be dropped, perceptually User 1 is unhappy about the dropping because the content is sensitive to the loss, but User 2 can tolerate even if a lot of frames are dropped because the content is not loss sensitive. With distortion fairness, User 1 can get more bandwidth than User 2, but perceptually both users are equally happy. Hence, bandwidth sharing should be content-aware and distortion fair, with the advantage that flows with less motion help motion-rich flows. Such advantage can be readily viewed through Fig. 3 from the motion-rich video "Foreman" (as User 1) and video "Akiyo" (as User 2), which is with less motions.

$$
\begin{aligned}
\text{minimize} \quad & \max D_s(\Lambda_s^l, l \in L(s)) \\
\text{subject to} \quad & \sum_{s \in S(l)} (R_s^0 - R(\Lambda_s^l)) \leq c_l, \, \forall l \\
& \Lambda_s^l \subseteq \Lambda_s^k, \, \forall s, \forall l, k \in L(s), k: \text{ the next hop of } l \\
& \text{variables } \Lambda_s^l, \forall s, \forall l \in L(s)
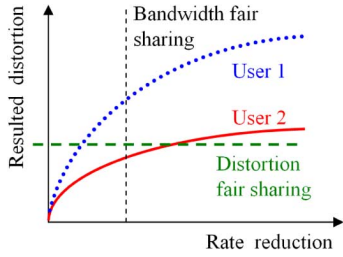\end{aligned}
\quad (8)
$$

Fig. 4. Illustration of distortion fair sharing.

Details of our scheme are presented in the next section.

IV. FRAME DROPPING FOR MAX-MIN UTILITY FAIR SHARE

When there is congestion on a link, we use the video traffic shaping technique discussed in Sections II and III-B to prune the video queues. Considering that frames may have different importance, we propose a smart frame dropping scheme.

A. Frame Dropping Scheme

The strategy is to send side information indicating the utility (importance) and length of each frame to the links, and then links and users cooperatively decide the optimal threshold of dropping frames for each user, where the frames with less importance than the threshold will be dropped. We define the utility of each frame based on the distortion induced if the frame is lost, while taking into account the prediction structures of the frames.

Min-max distortion fair (max-min utility fair) bandwidth allocation means at the optimal distortion, the distortion of User s cannot be decreased while still maintaining bandwidth feasibility, without increasing the distortion of some other user with a higher distortion than User s.

For the case of multiple users sharing only one link, the link decides a common threshold (distortion level) of dropping frames for all users, such that the total traffic through the link is less than the link capacity. The same threshold of dropping frames makes users experience a min-max fair distortion.

In a general network, the end-to-end path of a user may consist of several links, each of which shared by different users. If each link decides a common threshold of dropping frames for its users, since links are heterogeneous, on its end-to-end path, a user may see the links, each with a different threshold of dropping frames, and naturally the distortion level of dropping frames for a user should be decided by the most strict one, e.g., the highest distortion level over all the links on its path. If a link contains some user who is bottlenecked at other links, the link should know this user's frame dropping level, reduce its capacity by this user's flow, and re-allocate a common threshold of dropping frames for its other users with the remaining capacity. In this way, each user gets a distortion that is as low as possible, and the distortion of the user who experiences most stringent bottleneck is minimized, achieving the min-max distortion fair share.

B. Who Drop Frames: Links or Users?

In our approach, the users drop frames based on the threshold, which is found by the signaling over links and users in a control

TABLE I
COMPARISON OF DESIGNS

	Who drops	Content-awareness	Signaling
CBL	link	no	no
Baseline	link	simple	no
CAL	link	yes	no
CAF	user	yes	yes
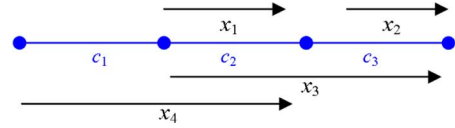
Fig. 5. Network with three links and four users.

TABLE II
COMPARISON OF USER DROPPING OVER LINK DROPPING

	x_1	x_2	x_3	x_4	y_1	y_2	y_3
Link dropping	180	100	100	180	200	540	200
User dropping	*200*	100	100	*200*	200	*500*	200

plane. In this way, the rate of the flows is controlled, and the link bandwidth is efficiently utilized.

The other option is that the links drop frames. The link can look at its buffer and drop the less important frames when it is congested. Such scheme looks fine if the bottlenecks are at the beginning of the end-to-end paths, but if the bottlenecks are in the middle or at the end of the paths, there will be large volume of the flow already over the links before the bottleneck link. Such scheme is not good in networks where bandwidth is very expensive, such as an edge network.

In Table I, some different designs are compared. Content-blind link dropping (CBL) is an approach that links acts as FIFO droptail queue and it is content-blind, not considering any frame information; baseline is an approach that links acts as FIFO droptail queue and it is partially content-blind where I frames are transmitted if room permits; content-aware link dropping (CAL) is an approach that links acts as a smart link where the frame dropping is content-aware, i.e., the link will find out the threshold of dropping frames based on the importance of every frame; and CAF is the approach proposed in this work that user drops frames based on importance, and the links cooperate and signaling is used to pass the massage, where the entire network including users and links is intelligent.

To see the advantage of user dropping versus link dropping, we present an example here. Consider a network as in Fig. 5. Assume $c_1, c_2, c_3 = 200, 540, 200$ kbps. Each user has a constant bit rate video with 200 kbps. Assume each user has the same utility function.

Table II shows the resulting transmitting rate x of users and the load y of links. User dropping saves bandwidth of link 2 by 8%, and rate of Users 1 and 4 increases by 11.1%.

User dropping frames not only provides better end-to-end quality (see Users 1 and 4), but also reduces the load of the links (see link 2). This is true for general network topology.

In addition, user dropping frames matches transmission control protocol (TCP) used in the Internet, which requires end users to adapt rate.

Since network link capacity may not have noticeable change in the scale of round trip time (RTT), in this work, we assume link capacities are constant for a short time interval.

C. Algorithm for the Optimal Threshold of Dropping Frames

We propose the following CAF algorithm to find the optimal threshold of dropping frames for each user.

CAF Algorithm

For a given interval W, the links and users do the following.

Initialization:

All links set "Done $= 0$" and $v_l = maxU$; all users set "$Done = 0$" and $z_s = maxU$, where $maxU$ is a given upper bound of the frame importance.

Iterations:

Do

   1) Threshold computation at link l:

  If link l "Done $= 0$"

     Link l reduces its capacity c_l by the rate (determined by z_s) used by users marked "$Done = 1$"

     Link l calculates a common threshold v_l for the remaining users within the remaining capacity.

     If link current v_l = previous v_l

     Link l set itself "$Done = 1$" and the link sets all its users "$Done = 1$" and $z_s = v_l$

     End If

  End If

   2) Threshold update at User s:

  If User s "$Done = 0$"

     User s set the frame dropping level $z_s = \max v_l$ over all the links on its path.

     Need a backward message passing.

  End If

Until

  All users' "$Done$"$= 1$.

User source adaptation:

User s drops the frames whose importance is less than z_s.

In the algorithm, each link finds an equal distortion level to its users not marked "Done"; each user sets its frame dropping level as the most stringent one over all the links on its path; afterwards, if the link is bottlenecked based on its users updated dropping level, the link marks itself and the users on it "Done", and other links reduce its bandwidth by the flow of the users marked "Done"; then the iteration goes to another round. The algorithm iterates among links and then flows, until the thresholds

cannot be reduced further. The criterion to decide whether a link is bottlenecked is that if the link is fully loaded or if the dropping level is zero at this link in the current iteration or, equivalently, if the current threshold would not change in the next iteration. Note that the concept of the bottleneck is extended to the bottleneck for the case that the flows are fully elastic, which means the flow can always increase if the link permits. In our case for the video clip which is already coded, the flow has a fixed maximum rate; hence, if the link has plenty of bandwidth which can support the maximum transmission rate of video, the frame dropping level is zero, but if the flow is fully elastic (no upper bound), the link may be fully loaded.

D. Number of Iterations of the Algorithm

In each iteration in CAF algorithm, there will be at least a link whose current threshold v_l will not change in the next iteration, and such link is a link which would be fully utilized if the flows on it are fully elastic without upper bound. Hence, we have the following Proposition.

Proposition 1: The number of iterations of the algorithm is at most the number of bottlenecks, where the bottleneck is counted in the same network where all the flows are fully elastic without upper bound.

V. SIMULATIONS

These simulation results are meant to demonstrate the effectiveness of the proposed scheme, not a full-scale deployment in a real network.

For the advantage of distortion-fairness over bandwidth-fairness, it can be readily viewed through Fig. 3 from the motion-rich video "Foreman" (as User 1) and video "Akiyo" (as User 2) which has less motion, as illustrated in Fig. 4. We omit the details here.

When the capacity at some links in the network is not sufficient to support the coming traffic, if we do not allow frame dropping, bottlenecks will appear at those links, with resulting delays creating playback "freeze ups" for certain flows. This is not desirable from end users' point of view. Instead, we allow certain frames to be dropped to clear up bottlenecks and to meet the stringent delay QoS requirements.

Obviously, most routers nowadays have packet drops in a content-blind fashion. When there is congestion, if the packet drops are content blind, an I frame could be dropped with resulting catastrophic loss of quality for the whole GoP. Due to the space limitation, we omit the comparison of CAF with these approaches. Instead, we show the comparison with the following more intelligent baseline approach.

A baseline approach for comparison is that the frames from users are served at each link in a FIFO fashion, and assume for time interval W (seconds) of one GoP, link l can only serve frames with total length of $W c_l$ (kbits) and the remaining frames in the GoP of the users on this link are lost. This also guarantees that the most important frame, I frame, in each GoP will be served if link capacity permits.

A. Two-Link Three-User Network

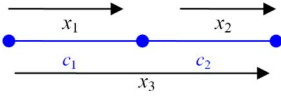First, we investigate a simple network with two links and three users, as shown in Fig. 6.
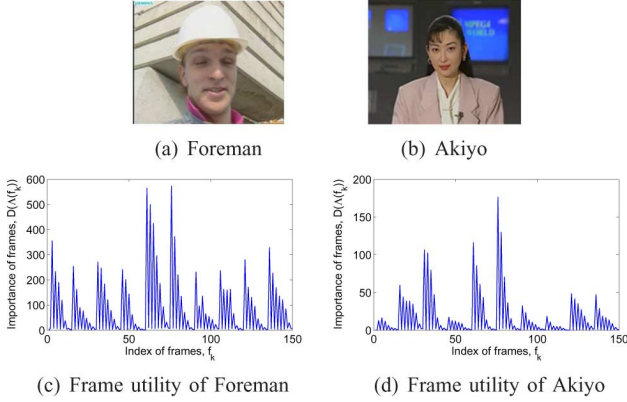
Fig. 6. Two-link three-user network.

(a) Foreman

(b) Akiyo

(c) Frame utility of Foreman

(d) Frame utility of Akiyo

Fig. 7. Video sequences and their utility of each frame.

(a) Distortion of user 1,3 after frame dropping.

(b) Queue length of link 1.
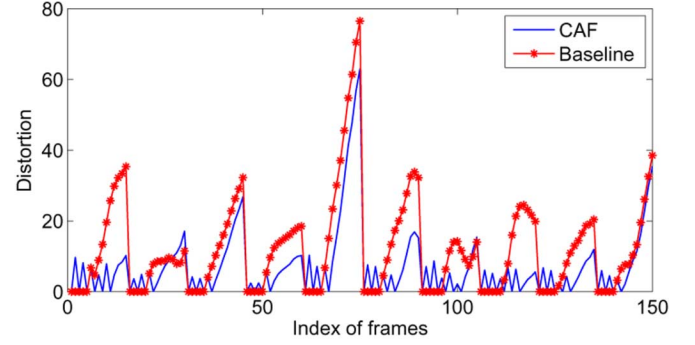
Fig. 8. Comparison of CAF and baseline.

Case a: Users 1,2,3 all send video clip Foreman.

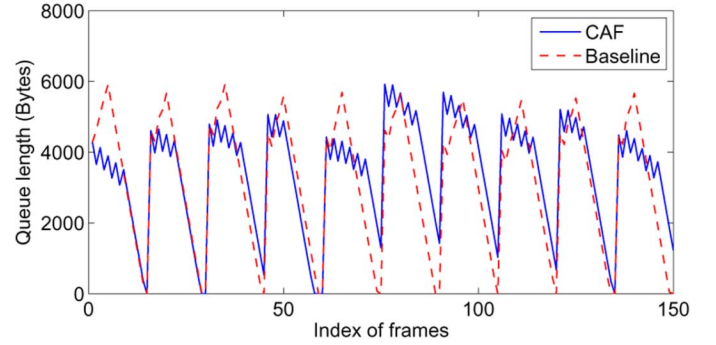Case b: Users 1,2 send Foreman, while User 3 sends Akiyo.

The link capacity: $c_1 = 150$ Kbps, $c_2 = 250$ Kbps. (Same results if c_1 and c_2 are swapped.) Video Foreman is encoded as variable bit rate (vbr) 144Kbps, and Akiyo is encoded as vbr 112Kbps. Each GoP has 15 frames, $IBPBPBP \ldots BP$. Fig. 7 shows the utility of each frame (taking into account the frame dependency) of the first 150 frames of the video. It can be seen that Akiyo has smaller distortion than Foreman because Akiyo has less motions.

For Case a, link 1 is congested and Users 1,3 have the same frame dropping. Link 2 is not congested and User 2 does not have frame dropping. We compare CAF with the baseline approach, where for CAF, the interval W is one GoP. Fig. 8(a) shows the perceptual distortion after frame dropping w.r.t. playback time. It can be seen that CAF approach can achieve much smaller distortion. Fig. 8(b) shows the queue length at the congested link 1. CAF and the baseline approach have the similar queue length, and each can stabilize the queue. Note that for the purpose of comparison, we align I frames of Users 1 and 3, but in practice, CAF does not need any alignment of users. The computing of threshold of dropping frames does not need any alignment and it can be done on-the-fly.

We compare Case a and b in Fig. 9 to show the advantage of CAF on content-aware distortion fair sharing. Fig. 9(a)–(c) shows the frame dropping, where the boundary is the frame-by-frame distortion $d(f_k, f_{k-1})$ without considering the frame dependency, which gives an indication of activity levels in the video sequence. The stems indicate the frames dropped. When two Foreman clips share link 1 as in Case a, the frames are dropped a lot due to the congestion. When Foreman and Akiyo share link 1 as in Case b, they have a common threshold of dropping frames because User 3 is bottlenecked by link 1. Since Akiyo does not have many motions, the distortion is smaller, a common threshold makes more frames dropped in Akiyo video clip, but Foreman video of User 1 benefits from less frame dropping. In this case, both User 1 and User 3 are happy about the

video quality. From Fig. 9(d), the fair utilities about the video quality can be seen.

In the following, we discuss how the interval of video for computing the threshold of dropping will affect CAF approach. For Case a, Fig. 10 shows that if the time interval W is larger, the distortion can be smaller. This is because a large W makes it possible for more frames to participate the cooperation which lets the important frames get through. A large W can also reduce the computation frequency for the threshold of dropping. But a large W may not track well the time varying characteristics of the video clips and it may not track the dynamics of user joining and leaving. In practice, W needs to be chosen considering these tradeoffs.

In Fig. 11, we show that CAF algorithm can work on-the-fly, when users come and go. We have the same setting as Case b, except that Users 1, 2, 3 transmit video sequence during the interval $[0, 5]$, $[2, 7]$, $[3, 8]$ s, respectively. The distortion after frame dropping is shown in Fig. 11. Users 1 and 3 have obvious frame drops in the interval $[2, 5]$ because of congestion.

For the same topology as in Fig. 6, but with different capacity, we show that the CAF algorithm, which is based on user dropping frame, performs better than the approach, where link drops frames. If link 1 is more congested than link 2, the CAL and CAF have the same performance because link 1 has shaped the flow without overshooting. But when link 2 is more congested, CAF can perform better than CAL in this case. Given $c_2 = 150$ kbps, for $c_1 = 210, 220, 230$ kbps, since link 2 is the congested link, link 2 is fully loaded in both CAF and CAL, but CAF save bandwidth of link 1 by 3.53%, 7.26%, 11.89% compared with CAL; User 2 and User 3 have the same performance
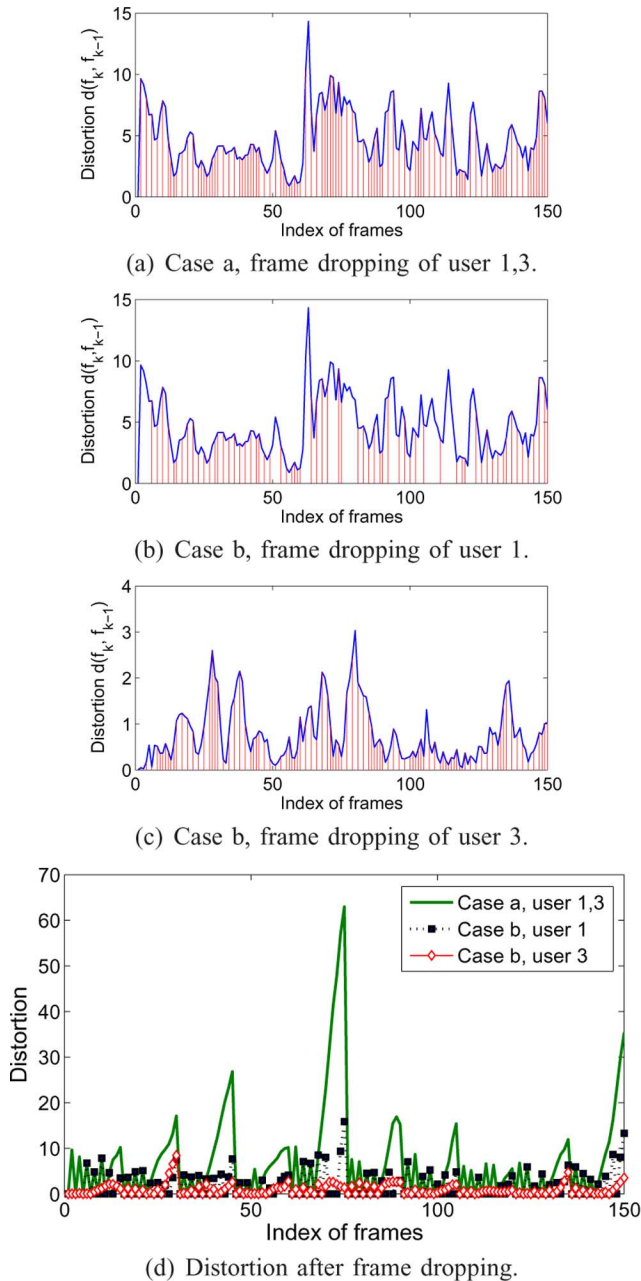
(a) Case a, frame dropping of user 1,3.

(b) Case b, frame dropping of user 1.

(c) Case b, frame dropping of user 3.

(d) Distortion after frame dropping.

Fig. 9. Illustration of content-aware distortion-fair sharing by an example of three users and two links.
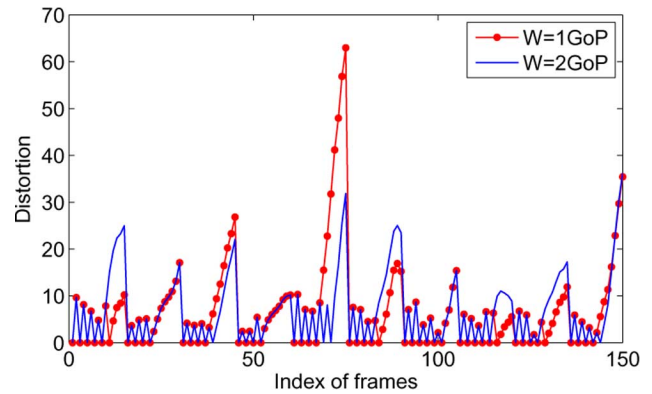

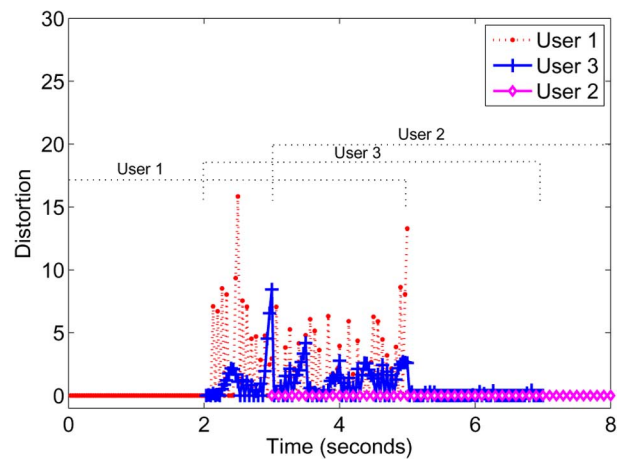Fig. 10. Comparison of different interval W. Figure shows the distortion after frame dropping.

Fig. 11. Distortion in the case of users' coming and leaving.

TABLE III
COMPARISON OF CAF AND CAL

c_1 (kbps)	210	220	230
Vs. CAL, CAF saves bandwidth	3.53%	7.26%	11.89%
User 1 distortion (CAL : CAF)	13.06 : 0	2.82 : 0	0 : 0

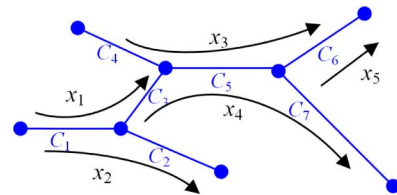Fig. 12. Seven-link five-user network. The link capacities are 150, 200, 250, 150, 200, 150, and 200 Kbps for links 1–7, respectively.

in CAF and CAL, but the distortion of User 1 by CAL is 13.06, 2.82, 0, while in CAF the distortion of User 1 is 0, 0, 0, respectively (see Table III). So CAF not only saves bandwidth but also provides better end-to-end distortion performance.

B. More Complex Network Topology

We tested CAF approach in more complex networks and our experiments demonstrate effectiveness of CAF. Here we show a typical example where the network has seven links and five users, as shown in Fig. 12. Users 1, 3 send video Foreman, Users 2, 4 send Akiyo, and User 5 sends Carphone (vbr, 144 Kbps). The utility of each frame taking into account of frame dependency for "Carphone" is shown in Fig. 13.

There are three bottlenecks in the network: links 1, 5, and 6. We verified that after three iterations of our algorithm, as indicated in the Proposition. Our CAF approach yields the distortion after frame dropping: for Users 1 and 2, they are the same as User 1 and User 3 in the previous Case b in Fig. 9(d) of the two-link three-user network; for Users 3 and 5, the distortion is shown in Fig. 14; and for User 4, it only has one B frame dropped and the distortion is almost zero. Note that Users 3 and 5 have almost the same distortion. Since User 3 is bottlenecked
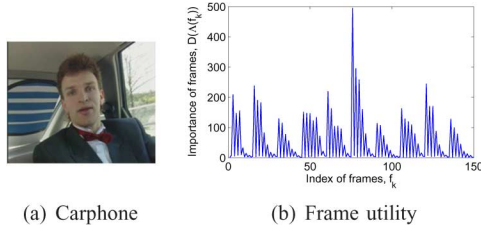
(a) Carphone   (b) Frame utility

Fig. 13.   Video clip Carphone, utility of each frame taking into account of frame dependency.
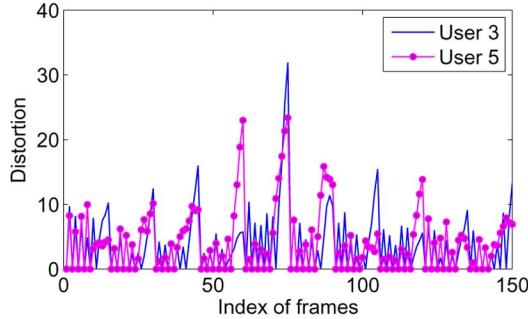
Fig. 14.   Distortion after frame dropping.

TABLE IV
MAX DISTORTION LEVEL

	User1		User2		User3		User4		User5	
	BL	CA	BL	CA	BL	CA	BL	CA	BL	CA
1	190	7.93	3.56	7.93	190	19.5	1.07	0	155	19.5
2	77.1	14.1	22.9	14.1	77.1	29.6	2.03	0	82.7	29.6
3	121	6.45	6.45	6.45	121	28.6	0.85	0	56.6	28.6
4	62.3	4.70	3.62	4.70	62.3	24.4	1.15	0	133	24.4
5	296	9.74	4.84	9.74	296	41.9	0.55	0	95.5	41.9
6	135	7.07	5.99	7.07	135	31.3	0.67	0	79.8	31.3
7	65.5	7.22	4.19	7.22	65.5	22.9	0.67	0	24.6	22.9
8	162	5.15	2.68	5.15	162	19.1	0.37	0	79.1	19.1
9	95.6	6.75	10.3	6.75	95.6	21.0	1.87	0	88.0	21.0
10	136	8.85	9.03	8.85	136	29.3	1.03	0.2	21.6	29.3

TABLE V
TOTAL DISTORTION

	User1	User2	User3	User4	User5
Baseline	1802	537.4	1802	83.03	1094
CAF	307.5	136.6	596.6	0.17	630.4

at link 6, its dropping at link 6 makes room for User 4 at link 5. We also examined the resulting video at the receiving ends, and the subject perception matches the distortion metric.

Table IV illustrates the performance in terms of the maximum distortion levels between baseline approach (BL) and CAF approach (CA). It can be seen that CAF achieves lower distortion level. Table V shows the total perceptual distortion of each user after frame dropping. In both cases, CAF outperforms baseline solution by a large margin.

Similar to the previous case illustrated in Table III, advantage of CAF over CAL can be shown. Due to the space limitation, details are omitted.

Note that CAF has limited computation and communication overhead. For stored video, the side information of frame length and importance is calculated in advance, so there is not much computational overhead for this. If CAF is used for real-time encoded video, it adds approximately 5% computation complexity to the encoder, as we found from our simulation. For communication overhead, it mainly comes from notifying the side information of frame importance and length to links. If we use 20 bits to send such information, the overhead is 0.6 kbps for each video stream of 30 frames per second. Within a given interval for CAF algorithm, as shown in the algorithm in Section IV-C, the computation and communication overhead is limited.

VI. CONCLUSION

In this paper, we propose a framework of CAF video networking for joint video source adaptation and network resource allocation, by combining the intelligence of the temporal scalability of the video source coding at the application layer with the congestion control at the transport layer, such that the min-max end-to-end video distortion fairness is achieved among users in a general multihop multiuser video delivery network.

We adopt the approach of users dropping less important frames when the network is congested. Architecturally we illustrate that the end users drop less important frames is in general more bandwidth-efficient than links drop less important frames when the network is congested.

Based on the framework, we propose an algorithm with a provable and small number of iterations for users and links to cooperate, to achieve the min-max distortion fairness. The application layer video streaming pruning and distortion metrics are utilized to guide transport layer resource allocation and coordination.

The advantages of our CAF approach include: 1) The dropping of frames when links are congested are content-aware and distortion-fair among all the users. 2) It provides the possibility that video clips help each other to get a fair quality-of-service (in the sense of perceptual distortion) when they share the resources (link bandwidth), i.e., lazy sequence helps out busy sequence. 3) Users drop frames based on computed thresholds, where the thresholds are obtained by the cooperations among users and links. Users' dropping frames is more bandwidth-efficient in general compared with links' dropping frames, which is very important for the networks whose bandwidth is expensive. 4) The threshold of dropping frame can be computed on-the-fly, based on the time-varying video content. There is no need to estimate a utility function which is hard to track the time-varying characteristics of video. Instead, we only need to signal the side information of the number of bits of the frames and the pre-computed frame importance to the links. 5) It can naturally handle random events such as user joining, leaving, etc.

The solution involves simple signaling between links and sources. Simulations demonstrated the effectiveness of this solution. Our approach has significant performance improvement compared with those without content-awareness, frame utility, or distortion fairness.

For future work, we may utilize more advanced coding features such as joint temporal-SNR scalability and metrics as well as multiple description coding schemes into the framework, and develop new solutions with more flexibility in content delivery network engineering, and higher end-to-end QoS for end users.

ACKNOWLEDGMENT

REFERENCES

[1] *MPEG-4 Visual,* ISO/IEC 14496-2.

[2] J. G. Apostolopoulos, T. Wong, W. Tan, and S. Wee, "On multiple description streaming with content delivery networks," in *Proc. IEEE INFOCOM*, Jun. 2002.

[3] P. Baccichet, J. Noh, E. Setton, and B. Girod, "Content-aware P2P video streaming with low latency," in *Proc. IEEE ICME*, Jul. 2007, p. 2.

[4] S. Bajaj, L. Breslau, and S. Shenker, "Uniform versus priority dropping for layered video," in *Proc. ACM SIGCOMM*, Oct. 1998.

[5] Z. Cao and E. W. Zegura, "Utility max-min: An application-oriented bandwidth allocation scheme," in *Proc. IEEE INFOCOM*, Mar. 1999.

[6] J. Chakareski, P. A. Chou, and B. Girod, "Rate-distortion optimized streaming from the edge of the network," in *Proc. Workshop Multimedia Signal Processing*, Dec. 2002.

[7] J. Chakareski and P. Frossard, "Distributed collaboration for enhanced sender-driven video streaming," *IEEE Trans. Multimedia*, vol. 10, no. 5, pp. 858–870, Aug. 2008.

[8] J.-C. Chen, S.-H. Chan, and V. Li, "Multipath routing for video delivery over bandwidth-limited networks," *IEEE J. Select. Areas Commun.*, vol. 22, no. 10, pp. 1920–1932, Dec. 2004.

[9] M. Chiang, "The content-pipe divide," in *Proc. Conf. Information Science and Systems*, Princeton, NJ, Mar. 2008.

[10] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition: A mathematical theory of network architectures," *Proc. IEEE*, vol. 95, no. 1, pp. 255–312, Jan. 2007.

[11] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 390–404, Apr. 2006.

[12] T. Cover and J. Thomas, *Elements of Information Theory*. New York: Wiley, 2006.

[13] B. Girod, M. Kalman, Y. Liang, and R. Zhang, "Advances in channel-adaptive video streaming," *IEEE Wireless Commun. Mobile Comput.*, vol. 2, no. 6, pp. 549–552, Sep. 2002.

[14] B. G. Haskell, *Digital Video: An Overview of MPEG-2,* 1997.

[15] R. Hormis, E. Linzer, and X. Wang, "Joint diversity- and rate-control for video transmission on multi-antenna channels," in *Proc. IEEE Globecom*, Nov. 2007.

[16] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, "Joint source adaptation and resource pricing for multi-user wireless video streaming," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 18, no. 5, pp. 582–595, May 2008.

[17] M. Kalman, P. Ramanathan, and B. Girod, "Rate-distortion optimized video streaming with multiple deadlines," in *Proc. IEEE ICIP*, Sep. 2003.

[18] H. Kanakia, P. P. Mishra, and A. R. Reibman, "An adaptive congestion control scheme for real time packet video transport," *IEEE/ACM Trans. Netw.*, vol. 3, no. 6, pp. 671–682, Dec. 1995.

[19] F. Kelly, A. Maulloo, and D. Tan, "Rate control for communication networks: Shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, Mar. 1998.

[20] Y. Li, Z. Li, M. Chiang, and A. R. Calderbank, "Content-aware video-quality-fair (CAF) streaming," in *Proc. IEEE Globecom*, Nov. 2008.

[21] Y. Li, A. Markopoulou, J. Apostolopoulos, and N. Bambos, "Content-aware playout and packet scheduling for video streaming over wireless links," *IEEE Trans. Multimedia*, vol. 10, no. 5, pp. 885–895, Aug. 2008.

[22] Z. Li, A. K. Katsaggelos, G. Schuster, and B. Gandhi, "Rate-distortion optimal video summary generation," *IEEE Trans. Image Process.*, vol. 14, no. 10, pp. 1550–1560, Oct. 2005.

[23] Z. Li, G. Schuster, and A. K. Katsaggelos, "MINMAX optimal video summarization and coding," *IEEE Trans. Circuits Syst. Video Technol.,* Special issue on Analysis and Understanding for Media Adaptation, vol. 15, no. 10, pp. 1245–1256, Oct. 2005.

[24] S. Low, "A duality model of TCP and queue management algorithms," *IEEE/ACM Trans. Netw.*, vol. 11, no. 4, pp. 525–536, Aug. 2003.

[25] S. H. Low and D. E. Lapsley, "Optimal flow control, I: Basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[26] N. Mastronarde and M. van der Schaar, "A queuing-theoretic approach to task scheduling and processor selection for video decoding applications," *IEEE Trans. Multimedia*, vol. 9, no. 7, pp. 1493–1507, Nov. 2007.

[27] J.-R. Ohm, "Advances in scalable video coding," *Proc. IEEE,* Special Issue on Video Coding, vol. 93, no. 1, pp. 42–56, Jan. 2005.

[28] D. Quaglia and J. C. de Martin, "Delivery of MPEG video streams with constant perceptual quality of service," in *Proc. IEEE ICME*, Aug. 2002.

[29] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.

[30] H. Seferoglu and A. Markopoulou, "Opportunistic network coding for video streaming over wireless," in *Proc. Packet Video*, 2007.

[31] H.-P. Shiang and M. v. d. Schaar, "Multi-user video streaming over multi-hop wireless networks: A distributed, cross-layer approach based on priority queuing," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 770–785, May 2007.

[32] G. J. Sullivan and T. Wiegand, "Video compression—from concepts to the H.264/AVC standard," *Proc. IEEE*, vol. 93, no. 1, pp. 18–31, Jan. 2005.

[33] W. Wang, M. Palaniswami, and S. H. Low, "Application-oriented flow control: Fundamentals, algorithms and fairness," *IEEE/ACM Trans. Netw.*, vol. 14, no. 6, pp. 1282–1291, Dec. 2006.

[34] D. Wu, S. Ci, and H. Wang, "Cross-layer optimization for video summary transmission over wireless networks," *IEEE J. Select. Areas Commun.*, vol. 25, no. 4, pp. 841–850, May 2007.

[35] D. Wu, Y. T. Hou, and Y.-Q. Zhang, "Transporting real-time video over the Internet: Challenges and approaches," *Proc. IEEE*, vol. 88, no. 12, pp. 1855–1875, Dec. 2000.

[36] F. Wu, S. Li, and Y.-Q. Zhang, "A framework for efficient progressive fine granular scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 11, no. 3, pp. 332–344, Mar. 2001.

[37] J. Xin, C. W. Lin, and M. T. Sun, "Digital video transcoding," *Proc. IEEE*, vol. 93, no. 1, pp. 84–97, Jan. 2005.

[38] F. Zhai, R. Berry, T. N. Pappas, and A. K. Katsaggelos, "A rate-distortion optimized error control scheme for scalable video streaming over the internet," in *Proc. IEEE ICME*, Jul. 2003.

[39] Q. Zhang and Y.-Q. Zhang, "Cross-layer design for QoS support in multihop wireless networks," *Proc. IEEE*, vol. 96, no. 1, pp. 67–76, Jan. 2008.

[40] Z. Zhang, Q. Sun, W.-C. Wong, J. Apostolopoulos, and S. Wee, "An optimized content-aware authentication scheme for streaming JPEG-2000 images over lossy networks," *IEEE Trans. Multimedia*, vol. 9, no. 2, pp. 320–331, Feb. 2007.

Ying Li (S'07) received the B.E. degree (with honors) and the M.E. degree in electrical engineering from Xi'an Jiaotong University, Xi'an, China, in 1997 and 2000 respectively, and the M.A. and Ph.D. degrees in electrical engineering at Princeton University, Princeton, NJ, in 2005 and 2008, respectively.

She is with Princeton University and has been working with Samsung Telecommunications America, Dallas, TX, since October 2008. She was a visiting Ph.D. student in Swiss Federal Institute of Technology (EPFL), Lausanne, Switzerland, in summer 2007 and in Motorola Multimedia Research Lab, Schaumburg, IL, in fall 2007, respectively. She worked as a faculty member of research and a teaching assistant in the Department of Information and Communication Engineering at Xi'an Jiaotong University from 2000 to 2003 and as a visiting scholar in Fuji Xerox Co. Ltd., Japan, from 2000 to 2001. Her research interests include communications, networking, optimization, information theory, and signal processing.

Zhu Li (M'01–SM'07) received the Ph.D. degree in electrical and computer engineering from Northwestern University, Evanston, IL, in 2004.

He was with the Multimedia Research Laboratory (MRL), Motorola Labs, Schaumburg from 2000–2008, where he was a Principal Staff Research Engineer. He is currently an Assistant Professor with the Department of Computing, Hong Kong Polytechnic University. His research interests include image/video analysis, manifold modeling and machine learning in biometrics and multimedia search, video coding and communication, and optimization and distributed computing techniques in multimedia networks and systems. He has three issued U.S. patents and more than 40 publications in book chapters, journals, and conference proceedings in these areas.

Dr. Li is the elected Vice Chair of the IEEE Multimedia Communication Technical Committee from 2008–2010. He received the Best Poster Paper Award at the IEEE International Conference on Multimedia and Expo (ICME) in 2006, and the DoCoMo Labs Innovative Paper Award (Best Paper) at the IEEE International Conference on Image Processing (ICIP) in 2007.

Mung Chiang (S'00–M'03–SM'08) received the B.S. degree (with honors) in electrical engineering and mathematics and the M.S. and Ph.D. degrees in electrical engineering from Stanford University, Stanford, CA, in 1999, 2000, and 2003, respectively.

He is an Associate Professor of electrical engineering and an Affiliated Faculty of Applied and Computational Mathematics and of Computer Science at Princeton University, Princeton, NJ. He was an Assistant Professor at Princeton University from 2003–2008. His research areas include optimization, distributed control, and stochastic analysis of communication networks, with applications to the Internet, wireless networks, broadband access networks, and content distribution.

Dr. Chiang's awards include Presidential Early Career Award for Scientists and Engineers 2008 from the White House, Young Investigator Award 2007 from ONR, TR35 Young Innovator Award 2007 from Technology Review, Young Researcher Award Runner-up 2004–2007 from Mathematical Programming Society, CAREER Award 2005 from NSF, as well as Frontiers of Engineering Symposium participant 2008 from NAE and Engineering Teaching Commendation 2007 from Princeton University. He was a Princeton University Howard B. Wentz Junior Faculty and a Hertz Foundation Fellow. His paper awards include ISI citation Fast Breaking Paper in Computer Science, IEEE INFOCOM Best Paper Finalist, and IEEE GLOBECOM Best Student Paper. His guest and associate editorial services include IEEE/ACM TRANSACTIONS ON NETWORKING, IEEE TRANSACTIONS ON INFORMATION THEORY, IEEE JOURNAL OF SELECTED AREAS OF COMMUNICATION, IEEE TRANSACTIONS ON COMMUNICATION, IEEE TRANSACTIONS ON WIRELESS COMMUNICATION, and *Journal of Optimization and Engineering*. He also has 16 patents filed and co-chaired the 38th Conference on Information Sciences and Systems.

A. Robert Calderbank (M'89–SM'97–F'98) received the B.Sc. degree in 1975 from Warwick University, Warwick, U.K., the M.Sc. degree in 1976 from Oxford University, Oxford, U.K., and the Ph.D. degree in 1980 from the California Institute of Technology, Pasadena, all in mathematics.

He is a Professor of electrical engineering and mathematics at Princeton University, Princeton, NJ, where he directs the Program in Applied and Computational Mathematics. He joined Bell Telephone Laboratories as a Member of Technical Staff in 1980 and retired from AT&T in 2003 as Vice President of Research. He has made significant contributions to a wide range of research areas, from algebraic coding theory and quantum computing to wireless communication and active sensing.

Dr. Calderbank served as Editor in Chief of the IEEE TRANSACTIONS ON INFORMATION THEORY from 1995 to 1998 and as an Associate Editor for *Coding Techniques* from 1986 to 1989. He was a member of the Board of Governors of the IEEE Information Theory Society from 1991 to 1996 and began a second term in 2006. He was honored by the IEEE Information Theory Prize Paper Award in 1995 for his work on the Z4 linearity of Kerdock and Preparata Codes (joint with A.R. Hammons, Jr., P.V. Kumar, N.J.A. Sloane, and P. Sole) and again in 1999 for the invention of space-time codes (joint with V. Tarokh and N. Seshadri). He received the 2006 IEEE Donald G. Fink Prize Paper Award and the IEEE Millennium Medal and was elected to the U.S. National Academy of Engineering in 2005.