Cross-Layer Optimization and Effective Airtime Estimation for Wireless Video Streaming

Mohammad A. Alsmirat and Nabil J. Sarhan
Department of Electrical and Computer Engineering
Wayne State University
Detroit, Michigan 48202
Email:{msmirat,nabil}@wayne.edu

***Abstract*—This paper develops a cross-layer optimization framework for video streaming from multiple sources to a central proxy station over a wireless network. The proposed framework manages the application rates and transmission opportunities of various video sources based on the dynamic network conditions in such a way that minimizes the overall distortion. The framework utilizes a novel online approach for estimating the effective airtime of the network. We demonstrate the effectiveness of the proposed framework and effective airtime estimation approach through extensive experiments.**

***Index Terms*—Bandwidth allocation, cross-layer optimization, effective airtime estimation, video streaming, wireless networks, WLAN.**

I. Introduction

The interest in video streaming over wireless networks has grown dramatically. This paper considers video streaming from multiple video sources (or stations) to a central station over a single-hop IEEE 802.11 wireless LAN (WLAN) network. This application is typical in video surveillance systems. As shown in Figure 1, the wireless video sources (video cameras or sensors) share the same medium and can be either battery-powered or outlet-powered. The central proxy station is connected with a high-bandwidth link to the access point, and thus this link is not deemed as a bottleneck in the system. Large systems may be composed of multiple such systems or cells.

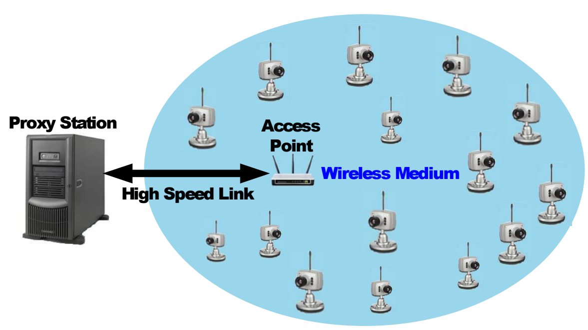


Fig. 1: A Wireless Video Surveillance System

The main challenge in the considered system is that the wireless network has limited available bandwidth, which should be estimated accurately and distributed efficiently among various video sources to maximize the quality of the video streams received by the proxy station. Bandwidth allocation in such systems has been addressed in only few studies [1], [2] by using cross-layer optimization. As discussed in Section II-B, these solutions are highly limited.

In this study, we propose a cross-layer optimization framework for managing the network bandwidth. The framework manages the application rates and transmission opportunities of various video sources based on the dynamic network conditions in such a way that minimizes the overall distortion. The framework utilizes a novel online approach for estimating the effective airtime of the network. To yield an accurate estimation, the proposed approach computes the effective airtime when the packet loss is below a specified threshold.

We evaluate the proposed solution by streaming real video frames (and not just an abstract bit stream) over a simulated network. The simulations are conducted using OPNET. Since the sent video packets may be lost, we implement an error concealment algorithm [3] at the proxy station to mitigate the impact of packet loss on perceptual video quality.

The main contributions of this paper can be summarized as follows. (1) We propose a complete cross-layer optimization framework, including a new online and dynamic approach for estimating the effective airtime of the network. (2) We develop a new and accurate model that characterizes the relationship between the video data rate and the distortion. (3) We develop a new model for adapting the link layer parameters in the video sources. (4) We test our framework by streaming real video frames over a simulated network and incorporating an error concealment algorithm to mitigate the impact of video packet loss.

The results show that the proposed framework enhances substantially the perceptual quality of received video streams and that the proposed effective airtime estimation algorithm is accurate and converges quickly. By sending and dropping much less data, the proposed framework results in much less power consumption, compared with existing solutions. Power consumption is a primary concern, especially when the video sources are battery-powered.

The rest of this paper is organized as follows. Section II provides background information and discusses the related work. Subsequently, Section III presents the proposed cross-layer optimization framework, and Section IV discusses the performance evaluation methodology. Finally, Section V

978-1-4673-1544-9/12/$31.00 ©2012 IEEE

presents and analyzes the main results.

II. Background Information and Related Work

A. IEEE 802.11e Standard

The 802.11e standard enables the provision of different quality-of-service (QoS) levels among different access categories (AC) in the same station, thereby enhancing the support of multimedia applications. These access categories include Voice, Video, Best Effort, and Background. The IEEE 802.11e MAC layer provides two methods for managing the access to the wireless channel: *Hybrid Coordination Function Controlled Channel Access* (HCCA) and *Enhanced Distributed Channel Access* (EDCA). In contrast with HCCA, EDCA provides reduced complexity and better flexibility by providing a distributed coordination function [2], [4].

With EDCA, priorities are implemented using four EDCA parameters: *Arbitration Inter Frame Space (AIFS)*, *Minimum Contention Window (CW_{min})*, *Maximum Contention Window (CW_{max})*, and *Transmission Opportunity Time (TXOP)*. AIFS controls the waiting time before an AC starts the transmission when the medium is not busy. In case of a collision, the AC will backoff for a random time between 0 and CW, where CW is a variable that is initialized to CW_{min}, is incremented after every transmission failure until it reaches CW_{max}, and is reset to CW_{min} after a successful transmission. The backoff timer is decremented every time the medium is sensed to be idle for at least $AIFS$ seconds. Finally, the TXOP limit controls the time period during which the AC keeps transmitting after it gains access to the medium.

B. Cross-Layer Optimization for Video Streaming

Numerous studies have discussed cross-layer optimization in video streaming over wireless networks. Studies [5], [6], [7] (and references within) consider a system in which only one station streams a video at a time, whereas studies [8], [9], [10] (and references within) consider a system in which multiple stations receive video streams form a central video server, and studies [1], [2] consider systems in which multiple stations deliver video streams to a central station. The latter studies are more related to this work.

Study [1] optimizes video streaming over 2G wireless network. The solution proposed in this study adapts the video streams by using video summarization techniques, such as frame skipping, which are not applicable to video surveillance because of the system's sensitivity for losing video frames.

Study [2] formulates and solves an optimization problem that minimizes the sum of distortion in all video streams. That paper used the formulation in [11] to develop an analytical model for the effective airtime. The model, however, is incorrect as will be discussed in Subsection II-C. In addition, that paper assumes p-persistent EDCA, which differs from the standard EDCA in the backoff timer selection process. Moreover, it ignored the packetization overhead of the transport and application layers when determining the optimal application rate and link layer parameters. In this paper, we address the problems of that study, and we also improve its link-layer adaptation model, which was based on the formulation in [12], [4].

C. Effective Airtime Estimation

The effective airtime is the fraction of the network time that is used for delivering useful data. As will be discussed later, solving the optimization problem requires an accurate estimation of the effective airtime. In [4], the effective airtime for ad-hoc networks was simply determined as the total throughput divided by the physical rate, assuming that all stations in the network have the same physical rate. Study [2] developed an analytical model for the effective airtime for video streaming from multiple stations to a proxy, based on the formulation in [11]. These two studies involve significant simplifications, approximations, and assumptions. The developed airtime model was simply given in terms of only CW_{min} and the number of stations in the network. Furthermore, according to the model, the effective airtime increases with the number of nodes and yields a value close to 1 $(i.e., 100\%)$ in networks with 30 stations or more. Such behavior is logically and empirically incorrect. As will be shown in Section V, this model leads to significant dropping after the optimization and gives relatively high distortion.

Other studies [13], [14], [15], [16], [17], [18] sought to determine other related parameters, such as the saturation bandwidth and network capacity for ad-hoc networks. None of these studies is directly applicable to finding the effective airtime in our considered system.

III. Proposed Cross-Layer Optimization Framework

This paper considers a system of video streaming from multiple video sources (or stations) to a central station over a single-hop IEEE 802.11 WLAN network in EDCA mode. As shown in Figure 1, the system has $S \geq 1$ video sources and each source s streams a different encoded video at rate R_s. Each video source s may have a different physical rate (y_s).

The ultimate goal of this study is to provide an optimal solution that dynamically distributes and allocates the available network bandwidth among various video sources. This solution should consider all system, video, network, and environmental aspects, which may change dynamically. Therefore, the proposed cross-layer optimization solution utilizes and dynamically controls parameters in three layers in the network stack: Application, Link, and Physical.

A. Cross-Layer Optimization Problem Formulation

As in [2], we formulate the problem as a cross-layer optimization problem of the sum of the distortion of all video streams received by the central proxy station. We follow the formulation in [2], but adapt it to include the packetization overhead of the transport and application layers. Since all video sources share the same medium, the bandwidth allocation solution should determine the fraction of airtime that each video source receives in the system. Obviously, the total airtime cannot exceed the effective airtime of the medium.

Specifically, the problem is formulated as: find the optimal fraction of the airtime allocation $F^* = \{f_s^* | s = 1, 2, 3, ..., S\}$ for various video sources that minimizes the total distortion ($\sum_{s=1}^{S} Distortion_s(r_s)$), where r_s is the application layer transfer rate for video source s, and S is the number of video sources. This optimization is subject to the following constraints. (1) The total airtime of all video sources is less than the effective airtime of the medium (A_{eff}). (2) The application layer transfer rate of source s is the product of the its airtime (f_s) and the physical layer transfer rate (y_s) for video source s. (3) The airtime of each source is between 0 and 1 (inclusive).

Mathematically, the problem can be stated as follows:

$$Find \quad F^* = \arg\min_{F} \sum_{s=1}^{S} Distortion(r_s) \tag{1a}$$

$$s.t. \quad \sum_{s=1}^{S} f_s = A_{eff} \tag{1b}$$

$$r_s = f_s \times y_s \tag{1c}$$

$$0 \leq f_s \leq 1 \tag{1d}$$

$$s = 1, 2, 3, ..., S, \tag{1e}$$

where F^* is the set of optimal fractions (f_s^*) of the airtime of all sources, r_s^* is the optimal application-layer rate of video source s, y_s is the physical-layer rate of video source s, and A_{eff} is the total effective airtime.

To solve the problem formulated in Equation (1), we need to characterize the distortion function and assess the effective airtime of the network.

B. Distortion Function Characterization

We seek to find a model of the relationship between the size of a JPEG snapshot (or alternatively the rate of an MJPEG video) and the distortion of the snapshot. We determine the size-distortion relationship based on the following image data sets: CMU/MIT [19], Georgia Tech [20], FERET [21], and SCFace [22]. For each image set, we use the IJG JPEG library to compress each picture in the set with quality factors from 1 to 100, with 1 being the lowest, and then assess the distortion of each image against the original image using the Root Mean Square Error (RMSE) metric. Finally, we find the average size and the average distortion of all images with the same quality factor. The results are shown in Figure 2. These results indicate that the model formulated in [23] (identified as "Existing Model") is not accurate. We determine that the distortion can be better characterized as follows:

$$Distortion(RMSE) = a \times Z^b + c, \tag{2}$$

where Z is the image size and a, b, c are constants. This model is referred to as "New Model" in Figure 2. For MJPEG videos, image size Z can be calculated as $Z = R/\tau$, where R is the video playback rate and τ is the video frame rate.

C. Effective Airtime Estimation

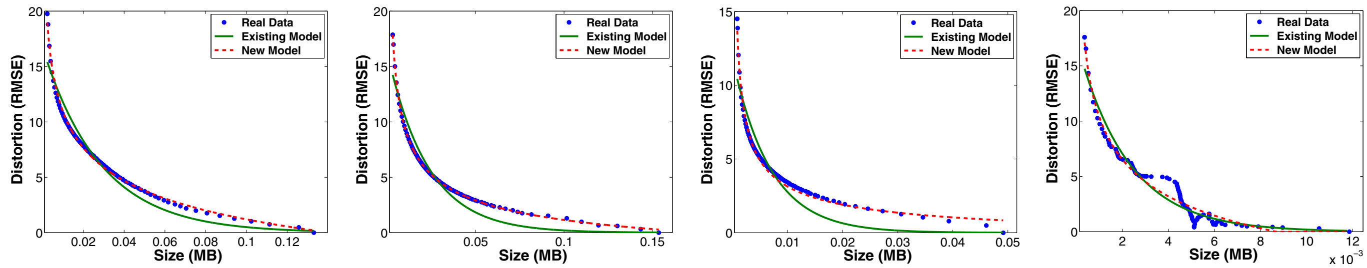
As finding an accurate value for the effective airtime of the medium is necessary for solving the formulated optimization problem, we propose a novel online and dynamic effective airtime estimation algorithm for wireless networks in infrastructure configuration. In contrast with existing analytical models, the algorithm uses complete information about the network and involves the cooperation of the access point and all video sources as well as various layers in each source. The algorithm does not incur additional network traffic and can be executed only when significant changes in the network (such as variations in the physical rates) happen.

As shown in Figure 3, the algorithm proceeds as follows. First, for each video source s, it finds the throughput (t_s) of its video stream as received by the application layer of the proxy station, when all sources stream videos, each at a rate that is equal to the maximum physical rate of the network divided by the number of sources [24]. The algorithm then uses this throughput to determine the initial value of the effective airtime (A_{eff}) as follows: $A_{eff} = \sum_{s=1}^{S} t_s/y_s$. Our experiments indicate that this initial value significantly overestimates the effective airtime, causing the actual received video rates to not conform to the cross-layer optimization solution because of the high level of packet packet dropping in the network. Thus, this value has to be adjusted based on the experienced level of packet dropping. Subsequently, during a period of time, called estimation period, each video source s assesses its own data dropping rate (d_s) while sending its video stream, and then sends this information to the access point (AP). Meanwhile, the AP determines the overall average dropping ratio as follows: $A_{\triangle} = \sum_{s=1}^{S} d_s/y_s$. $A_{\triangle}$ is used to adjust the current value of A_{eff} at the end of the current estimation period. If $A_{\triangle}$ is greater than some threshold A_{thresh}, the AP reduces A_{eff} by $A_{\triangle} - A_{thresh}$. A_{thresh} controls the allowable dropping in the network. If $A_{\triangle}$, however, is less than A_{thresh}, the AP increases A_{eff} by a value I, which set as half the last increment or decrement value, depending on whether the last operation is increment or decrement, respectively. Guided by extensive experiments, we set I to half the last decrement/increment to ensure better convergence and stability. The algorithm terminates when the increments become less than a threshold I_{thresh}. The estimation period and I_{thresh} should be chosen based on the best tradeoff between convergence and stability.

D. Cross-Layer Optimization Solution

Now that we have a distortion function and a value for the effective airtime, we can solve the problem formulated in Equation (1). The problem solution can be summarized as follows:

Step 1: We first prove that the formulated problem is a convex programming problem by determining that all constraints ((1b)-(1e)) in the problem are linear and thus convex and that the optimization function ($\sum_{s=1}^{S}(a_s(f_s y_s/\tau)^{b_s} + c_s)$) is also



(a) CMU/MIT (b) Georgia Tech (c) FERET (d) SCFace

Fig. 2: Size-Distortion Models

if this is the first time to run the algorithm
  $A_{eff} = \sum_{s=1}^{S} t_s/y_s$;
At the end of each estimation period{
  $A_{\Delta} = \sum_{s=1}^{S} d_s/y_s$;
  if ($A_{\Delta} < A_{thresh}$){
    if (last operation was decrement){
      $I = 0.5 * lastDecrement$;
      $A_{eff} = A_{eff} + I$;}
    else if (last operation was increment){
      $I = 0.5 * I$;
      $A_{eff} = A_{eff} + I$;}
    else //no decrements happened before
      //keep increasing A_{eff} to cause the first decrement
      $A_{eff} = A_{eff}$+ 0.05 ;
    if ($lastIncrement < I_{thresh}$)
      Stop the estimation algorithm;
  }
  else if ($A_{\Delta} \geq A_{thresh}$){
    $A_{eff} = A_{eff} - (A_{\Delta} - A_{thresh})$;
    $lastDecrement = A_{\Delta} - A_{thresh}$;}
}

Fig. 3: Simplified Algorithm for Dynamically Estimating the Effective Airtime

convex. The latter is valid since the derivative of the sum term is monotonically non-decreasing and convex.

Step 2: Since the problem is a budget constrained convex programming problem, it can be solved using the Lagrangian relaxation technique [25]. Thus, we can write the following Lagrangian-relaxed formula:

$$L(F^*, \lambda) = \sum_{s=1}^{S}(a_s(f_s y_s/\tau_s)^{b_s} + c_s) + \lambda(\sum_{s=1}^{S} f_s - A_{eff}), \quad (3)$$

where $0 \leq f_s \leq 1$, and $s = 1, 2, 3, ..., S$. Next, the Lagrangian conditions are formulated as follows:

$$\frac{\partial L}{\partial f_s} = 0 \text{ and } \frac{\partial L}{\partial \lambda} = 0. \quad (4)$$

Assuming that all video sources have the same b_s, which is empirically valid, solving these two equations yields the following solution:

$$f_s^* = (\frac{-\lambda^* \tau_s}{a_s b_s y_s (y_s/\tau_s)^{(b_s-1)}})^{(1/(b_s-1))}, \quad (5)$$

where

$$\lambda^* = (\frac{A_{eff}}{\sum_{s=1}^{S}(\frac{-\tau_s}{a_s b_s y_s (y_s/\tau_s)^{(b_s-1)}})^{(1/(b_s-1))}})^{(b_s-1)}. \quad (6)$$

E. Enforcing the Optimization Results

With the aforementioned solution, the AP determines λ^* and then each video source s determines its fraction of the airtime f_s^* after receiving λ^* from the AP. Subsequently, each video source s should change the application data rate, which is the video encoding rate in this case, as follows: $r_s^* = f_s^* \times y_s$. Finally, the link-layer parameters are determined based on the allocated airtime for each source. The link-layer parameters can either be the transmission opportunity duration limit (TXOP limit) or alternatively the frequency of the transmission opportunity. The control of the TXOP limit is more preferred since it is only one parameter, whereas the transmission frequency involves three parameters (AIFS, CW_{min}, and CW_{max}), which complicates the control process design.

In [2], the TXOP limit (x_s^*) for source s is determined as the time required to transmit the number of MAC data frames that source s can transmit during one beacon interval time (t_b). As discussed in Section V-C, our results show that choosing intervals other the beacon interval can achieve better performance. In particular, the received video quality improves with the chosen time interval up to a certain point, and then it starts to worsen. In addition, the time interval that achieves the best results varies with the network size. Furthermore, the model in [2] does not incorporate the packetization overhead of the transport and application layers.

In our framework, we address these problems as follows. Each video source determines its TXOP limit as the time required to transmit the packets that belong to a single video frame along with all associated overhead. Because of the cross-layer approach, the MAC layer in source s can know the frame rate τ_s of the video sent by the application layer in that source. The MAC layer can also determine the maximum load rate R_s coming from the source's upper layers. Using this information, we can determine the maximum video frame size L_s (with overhead) as R_s/τ and the number of MAC layer data frames per maximum video frame N_s as L_s/l_s, where l_s is the average data load per MAC layer data frame. Given that O_s is the average MAC and physical layers overhead, t_s is *Short Interframe Space (SIFS)* time, and t_a is the time required to send an acknowledgment, TXOP limit can be found as follows:

$$x_s^* = [\frac{L_s}{y_s}] + [\frac{O_s N_s}{y_s}] + [(2N_s - 1)t_s] + [N_s t_a], \tag{7}$$

where $\frac{L_s}{y_s}$ is the time required for transmitting the data of a single video frame and the overhead of the upper layers associated with that video frame, $\frac{O_s N_s}{y_s}$ is the time required for transmitting the associated MAC and physical layers overhead, $(2Ns - 1)t_s$ is the sum of the *SIFS* periods needed for transmitting all the packets of the video frame, and $N_s t_a$ is the time required for receiving all the acknowledgment packets of the video frame packets.

IV. Performance Evaluation Methodology

We use the OPNET simulator to conduct various experiments. In contrast with prior studies, which deal with video streams as abstract data streams with certain bitrates, we implement a traffic source in OPNET that streams MJPEG videos as Real-time Transport Protocol (RTP) packets. Moreover, we implement a realistic video streaming client at the application layer of the proxy station. This client receives and reassembles the RTP packets from various video streams, and then carries out error concealment to mitigate the impact of packet loss. The use of MJPEG enables the use of standard image data sets that are suitable for surveillance applications because the MJPEG video stream is a set of JPEG images. We use the CMU/MIT image database to assemble the video streams. The frames are chosen randomly from this database. The streamer at each source takes a bitrate, a frame rate, and an image set as inputs and produces a corresponding MJPEG video stream.

The implementation of the proposed framework is distributed between the video sources and the AP.

For comparison purposes, we implement the cross-layer algorithm proposed in [2], referred to here as *Distortion Optimization (DO)*. Our proposed solution is referred to as *Enhanced Distortion Optimization (EDO)*. We also compare the results with standard EDCA. In addition, we implement a variation of the standard EDCA, called *Adaptive EDCA*, in which the application layer in each video source adapts its video rate according to the physical rate of that source, and thus the rate is set as y_s/S.

Table I summarizes the main simulation parameters.

TABLE I: Summary of Simulation Parameters

Parameter	**Model/Value(s)**
Number of video sources	6, 10, 16, 20, 24, 28, 32
Simulation Time	10 min
Packet Size	1024 byte
Application Rate	Optimized, Default = Max Physical Rate / S
Video Frame Rate	20 frame/sec
Physical characteristics	Extended Rate (802.11g)
Physical Data Rate	Random from [18Mbpr 54Mbps]
Buffer size	256 Kbit
Video TXOP limit	Optimized, Default = $3008\mu s$
Video CW_{min}	15
Video CW_{max}	31
Video AIFS	2
Short Retry Limit	7
Long Retry Limit	4
Beacon Interval	0.02 second

V. Results Presentation and Analysis

A. Effectiveness of Using the Cross-Layer Approach in Bandwidth Allocation

Let us start by demonstrating the benefits of utilizing information from different network layers in bandwidth allocation. Figure 4 compares standard EDCA and adaptive EDCA in terms of perceptual video quality, as measured by the overall Peak Signal to Noise Ratio (PSNR). These results indicate that adapting the application rate according to the physical rate in each video source (as done in Adaptive EDCA) improves the PSNR by at least 50%!

B. Analysis of the Proposed Effective Airtime Estimation Algorithm

Extensive analysis of various design parameters in the effective airtime estimation algorithm indicates that their values are best set as follows to enhance the performance in terms of stability and convergence: $A_{thresh} = 0.0075$, $I_{thresh} = 0.0005$, and *Estimation Period* $= 5$ seconds.

Figure 5 shows that the effective airtime over the whole time of running the EDO solution. The algorithm converges quickly in all studied network sizes.

Let us now discuss the impact of A_{thresh}, which determines the allowable dropping in the network, on perceptual video quality. Figure 6 shows the results of running the proposed EDO solution for two different network sizes. The results show that the quality of the received video streams improves with A_{thresh} up to a point and then the quality starts to worsen. The peak happens when A_{thresh} is smaller than 0.01, suggesting that that optimal perceptual video quality is achieved when the dropping is very small.

C. Analysis of Link-Layer Adaptation

As discussed in Subsection III-E, study [2] determines the TXOP limit (x_s^*) for source s as the time required to transmit the number of MAC data frames that source s can transmit during one beacon interval time (t_b). Let us know discuss the impact of choosing time intervals other than the beacon interval. Table II shows the perceptual video quality results (measured in PSNR) when running the solution proposed in [2] (DO) for different time intervals and network sizes. These results indicate that perceptual video quality improves with increasing the chosen time interval until a peak is reached, and then it starts to worsen. Furthermore, the best value of the time interval varies with the network size.

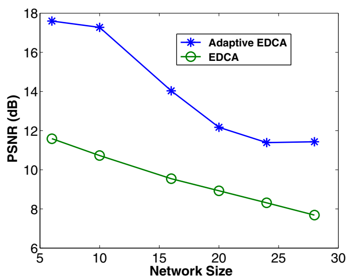


Fig. 4: Comparing EDCA with Adaptive EDCA

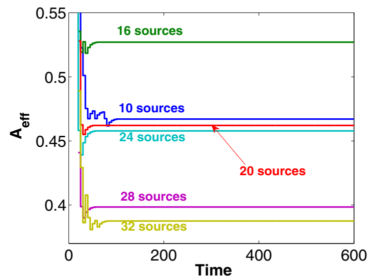


Fig. 5: Effective Airtime Estimation Progress

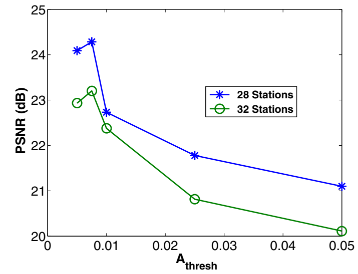


Fig. 6: Effect of A_{thresh} on PSNR [EDO]

TABLE II: Impact of the Time Interval Selected to Determine TXOP Limit on Perceptual Video Quality [PSNR (dB)]

Time Interval \ Network Size	**10**	**16**	**20**	**24**
0.02	19.15	12.30	10.25	8.81
0.5	24.46	18.77	16.17	15.69
1	24.46	19.00	16.40	15.66
2	24.46	19.00	16.46	15.66
3	24.46	19.00	16.46	15.65

D. Comparing Various Bandwidth Allocation Solutions

Figure 7 compare the performance of various solutions (EDCA, DO, EDO) in terms of the following metrics: (a) perceptual video quality, (b) average video packet delay, (c) overall data rate received by the application layer of the proxy from all video sources, (d) percentage of received complete video frames, (e) percentage of received incomplete video frames, (f) percentage of missed video frames, (g) overall network load (defined as the total load sent from the application layers of all sources), (h) overall dropping rate due to buffer overflow, and (i) overall dropping rate due to reaching the retransmission limit.

The perceptual video quality is the main metric and is measured in PSNR. This metric also considers the impacts of all other metrics. We set the PSNR of missed frames to 0. The results show that the proposed solution (EDO) significantly outperforms exiting solutions. In particular, it improves PSNR by more than 100% compared with standard EDCA and by 20% to 100% compared with DO. Interestingly, the overall received data rate is not always consistent with perceptual video quality. The average video packet delay is an essential metric due to the real-time playback requirement. The result show that EDO reduces the average packet delay significantly compared with EDCA and DO. In addition, EDO yields the highest percentage of completely received video frames and the least percentage of missed frames. This behavior is due to the effective airtime estimation algorithm, which minimizes packet dropping and to the effective link-layer adaptation model used. Accordingly, the EDO reduces significantly the last three metrics: overall network load, buffer dropping rate, and retransmission dropping rate. These results indicate that EDO consumes much less power by sending and dropping much less data.

VI. Conclusions

We have proposed a cross-layer video optimization framework that manages the network bandwidth to minimize the total distortion in video streams. The proposed framework utilizes a new online network effective airtime estimation algorithm. Moreover, we have developed a new and accurate model for characterizing the video data rate and distortion relationship as well as a new model for adapting the link-layer parameters. We have evaluated our framework by streaming real video frames over an OPNET-simulated wireless network.

The main results can be summarized as follows. (1) The proposed framework enhances substantially the perceptual quality of received video streams. (2) The proposed effective airtime estimation algorithm is accurate and converges quickly. (3) Optimal perceptual video quality is achieved when the packet dropping is very small. (4) The transmission opportunity adaptation model works effectively. (5) The proposed framework results in much less power consumption, compared with existing solutions. This behavior is due to sending and dropping much less data. Power consumption is a primary concern, especially when the video sources are battery-powered.

References

[1] J. Huang, Z. Li, M. Chiang, and A. K. Katsaggelos, "Pricing-based rate control and joint packet scheduling for multi-user wireless uplink video streaming," in *Proc. 15th International Packet Video Workshop (PV2006)*, 2006.

[2] C. Hsu and M. Hefeeda, "A framework for cross-layer optimization of video streaming in wireless networks," pp. 5:1–5:28, February 2011.

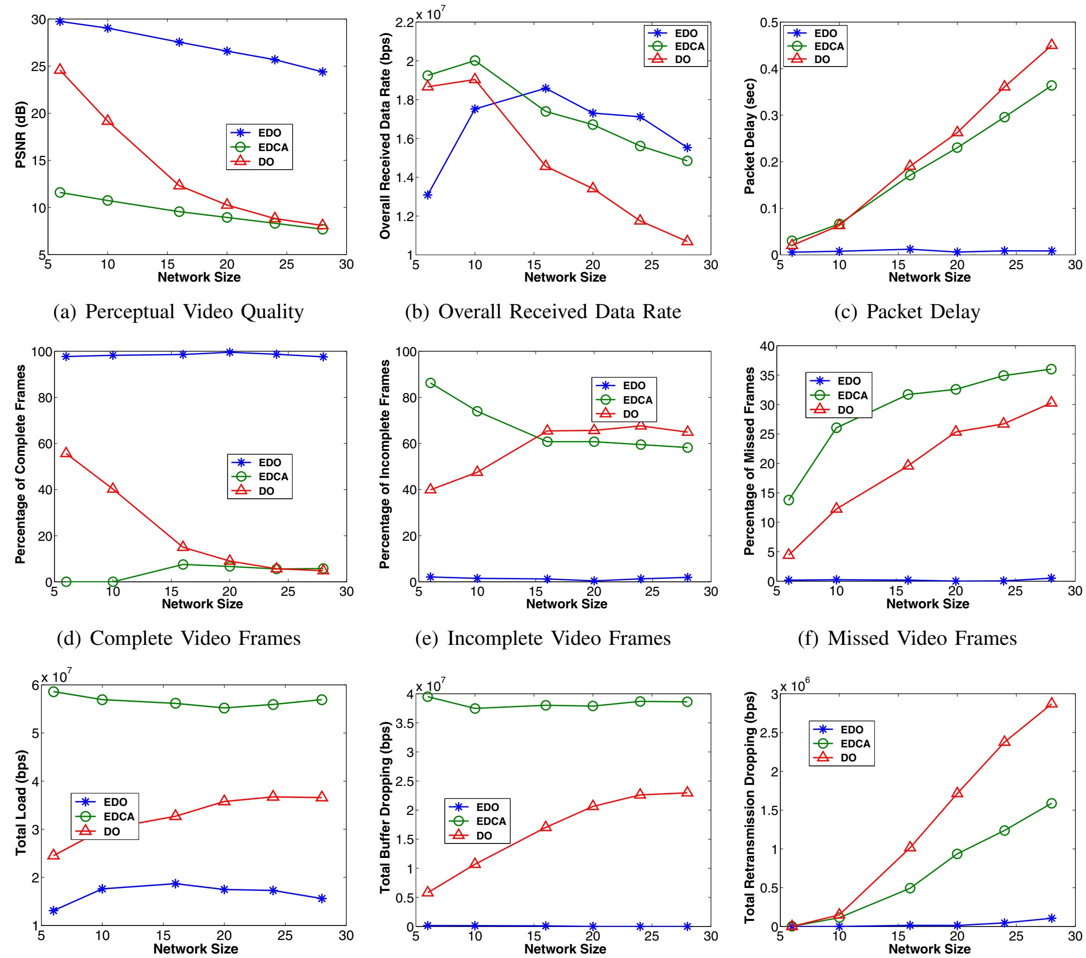
[3] S. Shirani, F. Kossentini, S. Kallel, and R. Ward, "Reconstruction of jpeg coded images in lossy packet networks."

[4] N. Shankar and M. van der Schaar, "Performance analysis of video transmission over ieee 802.11a/e wlans," *IEEE Transactions on Vehicular Technology*, vol. 56, no. 4, pp. 2346 –2362, July 2007.

[5] X. Lu, E. Erkip, Y. Wang, and D. Goodman, "Power efficient multimedia communication over wireless channels," *IEEE Journal on Selected Areas in Communications*, vol. 21, pp. 1738–1751, 2003.

[6] Z. He and D. Wu, "Resource allocation and performance analysis of wireless video sensors," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, pp. 590–599, 2006.

[7] Y. Andreopoulos, N. Mastronarde, and M. van der Schaar, "Cross-Layer Optimized Video Streaming Over Wireless Multihop Mesh Networks," *Selected Areas in Communications, IEEE Journal on*, no. 11, pp. 2104–2115.



(a) Perceptual Video Quality (b) Overall Received Data Rate (c) Packet Delay

(d) Complete Video Frames (e) Incomplete Video Frames (f) Missed Video Frames

(g) Overall Network Load (h) Overall Buffer Dropping Rate (i) Overall Retransmission Dropping Rate

Fig. 7: Comparing Various Bandwidth Allocation Solutions

[8] S. Khan, J. Brehmer, W. Kellerer, W. Utschick, and E. Steinbach, "Application-driven cross-layer optimization for video streaming over wireless networks," *IEEE Communications Magazine*, vol. 44, pp. 122–130, 2006.

[9] H. Zhang, Y. Zheng, M. A. Khojastepour, and S. Rangarajan, "Cross-layer optimization for streaming scalable video over fading wireless networks," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 3, pp. 344–353, 2010.

[10] 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 Transactions on Mobile Computing*, vol. 5, pp. 755–768, June 2006.

[11] Y. Ge, J. C. Hou, and S. Choi, "An analytic study of tuning systems parameters in ieee 802.11e enhanced distributed channel access," *Computer Networks*, vol. 51, no. 8, pp. 1955–1980, 2007.

[12] C.-T. Chou, S. N. Shankar, and K. G. Shin, "Achieving per-stream qos with distributed airtime allocation and admission control in ieee 802.11e wireless lans," in *INFOCOM*, 2005, pp. 1584–1595.

[13] J. Hui and M. Devetsikiotis, "A unified model for the performance analysis of IEEE 802.11e EDCA," *Communications, IEEE Transactions on*, no. 9, pp. 1498–1510.

[14] L. Xiong and G. Mao, "Saturated throughput analysis of IEEE 802.11e EDCA," *Computer Networks*, vol. 51, no. 11, pp. 3047–3068, 2007.

[15] Z. Yang and K. Nahrstedt, "A bandwidth management framework for wireless camera array," in *NOSSDAV*, 2005, pp. 147–152.

[16] S. H. Shah, K. Chen, and K. Nahrstedt, "Dynamic bandwidth management in single-hop ad hoc wireless networks," *MONET*, vol. 10, no. 1-2, pp. 199–217, 2005.

[17] S. H. Shah, K. Chen, K. Nahrstedt, and I. Introduction, "Available bandwidth estimation in ieee 802.11-based wireless networks," 2003.

[18] C. Sarr, C. Chaudet, G. Chelius, and I. G. Lassous, "Bandwidth estimation for ieee 802.11-based ad hoc networks," *IEEE Trans. Mob. Comput.*, vol. 7, no. 10, pp. 1228–1241, 2008.

[19] "CMU/MIT Image Set," http://vasc.ri.cmu.edu/idb/html/face/frontal_images.

[20] "Georgia Tech Face Database," http://www.anefian.com/research/face_reco.htm.

[21] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET database and evaluation procedure for face-recognition algorithms," *Image Vision Comput.*, vol. 16, no. 5, pp. 295–306, 1998.

[22] M. Grgic, K. Delac, and S. Grgic, "Scface - surveillance cameras face database," *Multimedia Tools Appl.*, vol. 51, no. 3, pp. 863–879, 2011.

[23] W. Cheng, X. Chen, and Z. He, "Doubling of the operational lifetime of portable video communication devices using power-rate-distortion analysis and control," in *ICIP*, 2006, pp. 2473–2476.

[24] A. Symington and P. S. Kritzinger, "A hardware test bed for measuring ieee 802.11g distribution coordination function performance," in *MASCOTS*, 2009, pp. 1–7.

[25] A. Ortega and K. Ramchandran, "Rate-distortion methods for image and video compression," *IEEE Signal Processing Magazine*, vol. 15, no. 6, pp. 23–50, 1998.