A Cross-Layer Framework for Adaptive Video Streaming over IEEE 802.11 Wireless Networks

Santhosha Rao[1], M. Vijaykumar[1], and Kumara Shama[2]

[1] Department of Information and Communication Technology
[2] Department of Electronics and Communication Engineering,
Manipal Institute of Technology, Manipal University, India
msanthosharao@gmail.com, vijayns02@gmail.com,
shama.kumar@manipal.edu

Abstract. Development in video compression techniques and wireless techniques has resulted in considerable amount of video streaming applications in the wireless networks. However, the time varying transmission characteristic of the wireless channels leads to poor performance of multimedia traffic over wireless networks. This results in longer packet delay, jitter and lower throughput that deteriorate the video quality considerably at the receiving end, thus diminishing the user experience. In this work, we propose cross-layer framework which optimizes the transcoding rate at the application layer depending upon the channel condition estimated using parameters associated with the data-link layer. Further, we evaluate the proposed frame work using NS-2 simulator with EvalVid framework. We use three different motion video sequences to evaluate the proposed frame work. Our simulation result shows that the proposed cross-layer frame work improves the video quality in all the three cases at the receiving end.

Keywords: adaptive video streaming, cross-layer, wireless networks.

1 Introduction

The use of the Wireless Local Area Networks in public areas, offices and homes has been scattering quickly and connection with internet is also increasing. Traffic carried by wireless networks is expected to be real time traffic such as video-on-demand, video conference and voice. To provide the quality video over wireless networks is challenging , due to the time-varying and erratic nature of a wireless channel and the strict delivery requirements of media traffic. Time varying transmission characteristic of the wireless channel leads to packet delay, packet loss, jitter and lower throughput. In video streaming, to set the play-out buffer at the receiving end is to reduce the jitter effect and smooth out the video played by the video client. However, additional packets are dropped by the play-out buffer because the important packet was lost or arrived later than allowable time. The traditional layered approach addresses this problem by implementing the prediction and adaptation algorithm at the lower layers of the protocol stack, specifically the transport, the medium access and the physical

N. Meghanathan et al. (Eds.): CCSIT 2011, Part I, CCIS 131, pp. 80–89, 2011.

layers. The lower layers are optimized independently without considering the effect on the other layers. These approaches do not adaptively support the real-time traffic such as the video streaming.

To increase the video quality at end user, we require an adaptive encoder, which considers the network conditions and varies its encoding rate accordingly. In order to achieve this, we require a feedback mechanism from lower layer which the existing layered architecture does not support. A cross-layer design is required to provide the necessary feedback from the lower layers. The concept of cross-layer design is based on inter-layer information exchange across the layers which aims at achieving mutual optimization of two or more layers. Even though this concept can be used in all communication networks, it is particularly appropriate in wireless networks because of the unique challenges concerned with the wireless environment. In this work, we use the layered architecture along with cross-layer information exchange to increase the video quality at end user.

Based on application requirements, different solutions to vary the video rate are proposed in literature. The rate-distortion optimized frame work is introduced in [1]. According to this framework, the packets are dropped intelligently based on their relative importance and priority. In certain applications the video encoder can adapt to the channel conditions by changing the compression degree and thus modifying the video rate [2]. The video encoder is able to get the channel quality by receiving the feedback information from the receiver side. However, such a proposal is hopeless when the round-trip delays are long. In [3] the authors pre-encoded video data at different rates. The end user selects the desired rate based on his requirement, which is not an appropriate approach for live video streaming. To mitigate the above mentioned problems we use the real-time transcoder for live video streaming in our approach.

We use retransmission information from data-link layer to estimate the network condition and vary the transcoder compression rate accordingly. In order to achieve this, we propose the cross-layer frame work which captures the retransmission information from data-link layer and feeds this information to the Application layer. The remaining part of the paper is organized as follows. Section 2 describes the proposed cross-layer frame work. Section 3 describes simulation set up and analysis of the result and section 4 provides a brief conclusion about our work.

2 The Proposed Cross Layer Framework

We considered IEEE 802.11 infrastructure mode wireless network and scenario where a video is streamed from a video server to base station through internet. The video from the video server is first received at the access point before being delivered to the appropriate wireless node. Therefore, the proposed cross layer framework is implemented in access point (AP). As shown in Fig.1 the frame work consists of the optimizer block along with the input/output interface to the protocol stack. We explain in detail, the optimizer block and its interfaces in the next subsections.

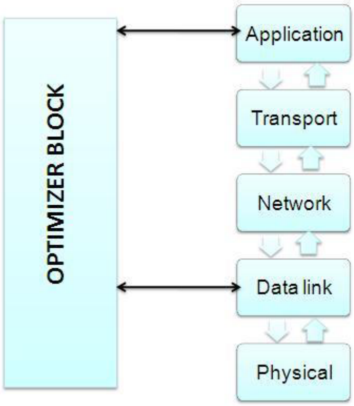

Fig. 1. Protocol stack for video streaming optimization

2.1 Input

The input to the optimizer block is obtained from the data-link layer. In data-link layer, information of packet transmission attempt is maintained in a counter. The data-link layer repeatedly transmits a packet until it receives a positive acknowledgment. Therefore, multiple attempts can be made to achieve successful transmission. Thus, the packet transmission attempts made at the data-link layer act as input data for the optimizer block.

2.2 Optimizer Block

Optimizer Block is similar to channel estimator. Optimizer will get the input from the data-link layer, based on this information it will estimate the channel condition. When input counter value is one, it means that there is no delay in network, thus indicating that the channel condition is good. When input counter value is two, it indicates that the channel condition is moderate. When input counter value is three or four, it indicates that the channel condition is bad. Based on the available channel information from the optimizer, the application layer then invokes the optimal approach to transcode the incoming video bit-stream.

2.3 Output

Optimizer block output is fed to the transcoder rate controller which will give the input parameter value to the transcoder. Based on this parameter, transcoder varies its encoding rate. When channel condition is good, it encodes the incoming video data at higher rate. When channel condition is moderate, it will encode the incoming video data at moderate rate and when channel condition is bad, it will encode the incoming video data at lower rate.

2.4 Video Transcoding Rate Calculation

Video compression is achieved by reducing the amount of data that is required to send. The main factor affecting the spatial detail is the quantization parameter. Details of the video are preserved when the Quantization Parameter (QP) is small. Some of the spatial detail is reduced when the QP is raised, leading to the decrease in bit rate. Hence, variation in the QP causes the variation in the bit rate. To control the transcoder mainly we require two parameters: i) target bit rate ii) QP. By default, transcoder recompresses the incoming frame at the rate of R (bits/frame). A transcoder is said to be an adaptable transcoder when it varies its transcoding rate according to the network conditions. When channel condition is good, we increase the transcoder target rate to 1.2 R (R_{max}) for higher video quality. When channel condition is moderate, transcoder target bit rate will be same as it is, to take the full advantage of the channel. When channel condition is bad, we decrease the transcoder target bit rate to 0.8R (R_{min}), to reduce the retransmission, packet loss at end user, smooth-out the transcoded video stream and also reduce the load on the network. Here, we select the rate alteration factor of 1.2 because higher values lead to disorder of the pre-calculated bit-budget distribution in H.264 encoder [4], which should be avoided. Furthermore, in case of bad channel condition, we use the rate alteration of 0.8 because lower values change the video quality. To choose the QP when the target bit-rate is available, both the linear and quadratic rate quantization (RQ) models are suggested. The quadratic RQ model is not appropriate for real-time transcoding because of its complexity [5] in QP selection, even though the quadratic model has higher accuracy than the linear model. Accordingly, a linear RQ model [6] is chosen in this work to get a better balance in the tradeoff between accuracy and complexity for real-time video streaming application.

Summary: In Fig.2, flow chart represents the procedure used in the proposed cross layer framework. The first frame is fed to the transcoder and its target rate is initialized to R_{max}. If the recent frame is not the first frame, then the transcoder computes the new transcoding QP value, based on the following information: available bandwidth, bit rate of the previous rate, frame rate and channel information from the optimizer block. If the incoming frame to transcoder is B frame, the calculated QP is used for transcoding. Otherwise, in case of I and P frames, previous calculated QP is used for transcoding. Optimizer block estimates the channel condition based on the number of transmission attempts for a packet. The channel condition information is fed back to the application layer for transcoding the next video frame.

3 Simulation Setup

We evaluate proposed cross layer framework using the NS-2 [8] network simulator, over IEEE 802.11 wireless network. CBR and FTP applications can be simulated using built in NS-2 agents, but there is no agent for video traffic application in NS-2. So, to achieve simulation of the video traffic, we use the EvalVid framework [9] and the JM reference software [7] to transcode the video sequences.

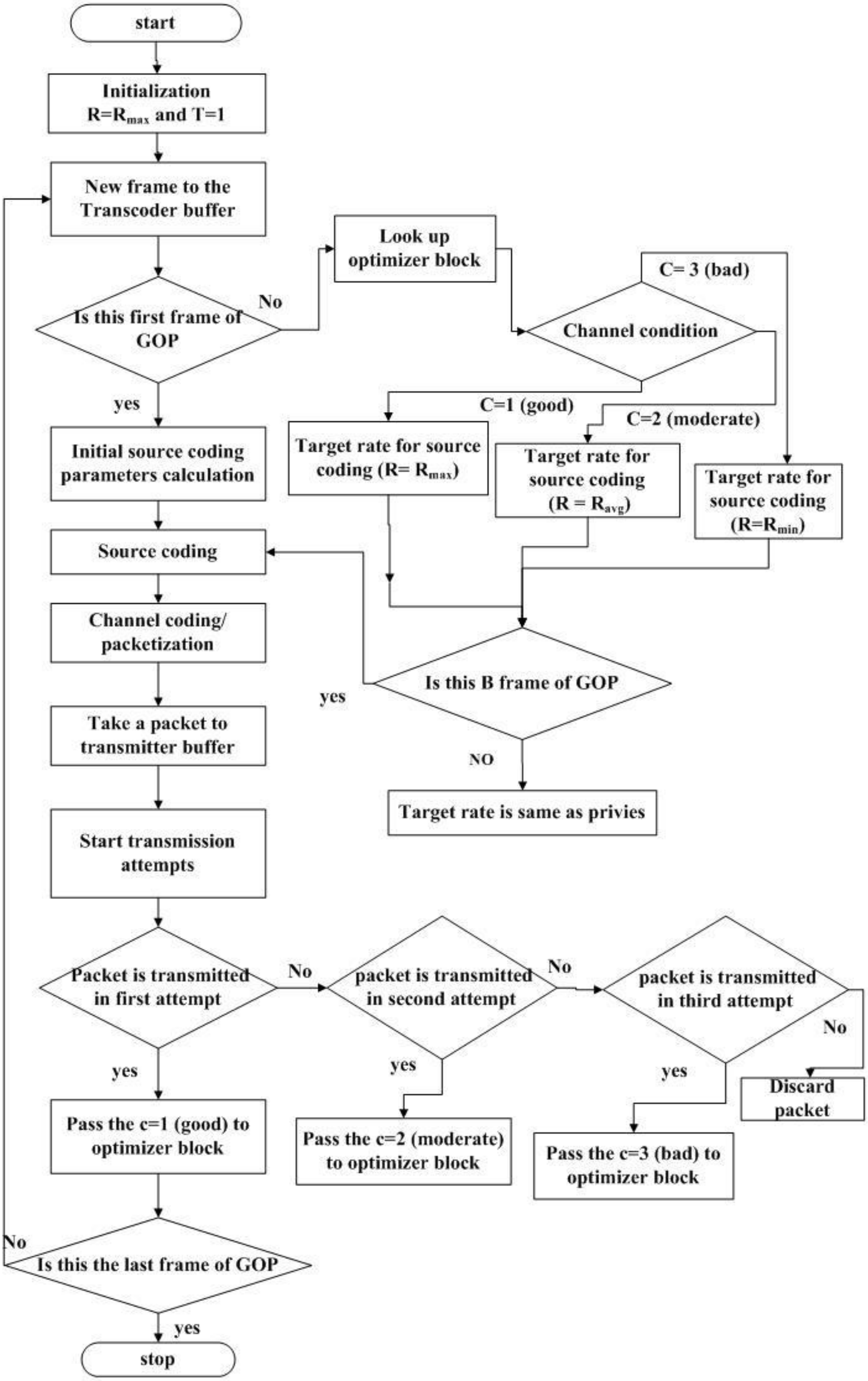

Fig. 2. Flow of operations in the proposed cross-layer framework

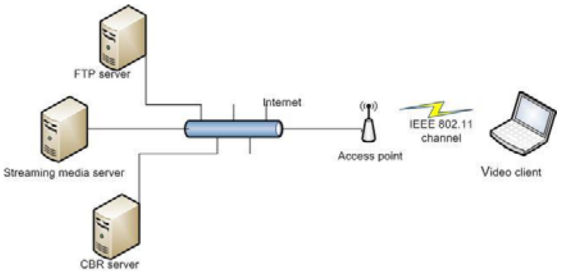

Fig. 3. Simulation model

To create realistic network conditions, we consider simulation environment as in Fig.2. The network traffic between the server and access point involves traffic due to FTP, CBR and the video data which is of our interest. FTP traffic is transmitted over TCP protocol and CBR traffic is transmitted over UDP protocol. The video traffic is transmitted over the myudp protocol. This protocol is similar to udp protocol. We use the myudp protocol to handle the video traffic in the simulation. The FTP source represents a bulk file transfer application with a maximum packet size of 1000 bytes and source rate is set at 512 kbps. The file considered is enough such that there is always data to transmit over the entire duration of the simulation. CBR represents the bursty traffic, with a maximum packet size of 1280 bytes and the source rate is set at 256 Kbps. In this simulation, we use mother-daughter, News and Foreman video sequences, which are example of the slow, medium and fast motion video clips respectively. These video clips are used as video traffic in simulation and the first 250 frames of each of these video clips are encoded using H.264 encoder. The first frame of the video sequence is an I-frame, while the subsequent frames are P-frames and B-frames. The video frame rate is set to 30 frames per second. Context Adaptive Binary Arithmetic Coding (CABAC) [4] was used for the entropy encoding and RD optimization [4] was enabled. Each video frame is segmented into small packets in transmission and the maximum packet size was set to 1000 bytes. The link between the base station and the video receiver is IEEE 802.11b 2Mbps link. For simplicity, we assume that the link between the base station and the video server has 10Mbps bandwidth and 10 ms latency. The Simulation results are analyzed in next section.

3.1 Results

Objective quality: Peak-signal-to-noise-ratio (PSNR) is one of the most well known objective metrics to evaluate the application level Quality of Services (QoS) of video transmissions and is therefore used in this work to measure video quality. PSNR measures the error between the encoded video at sender side and decoded video at receiver side. Fig.4, Fig.5 and Fig.6 show the improvement in PSNR with the proposed cross-layer frame work for the test video sequences Mother-Daughter, News, and Foreman respectively. Two different scenarios are considered to compare the video quality at end user. (1) Application layer parameters is adapted considering the channel conditions (denoted by with cross layer), (2) without any adaptation of

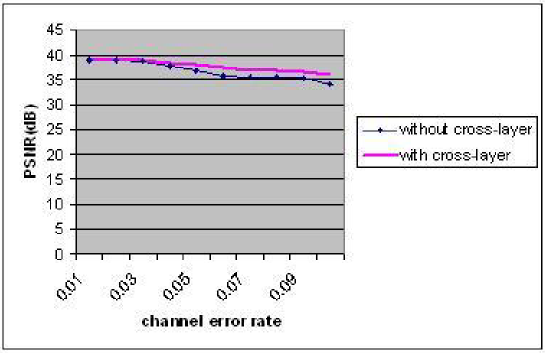

Fig. 4. Objective video quality of the Mother-daughter test video sequence

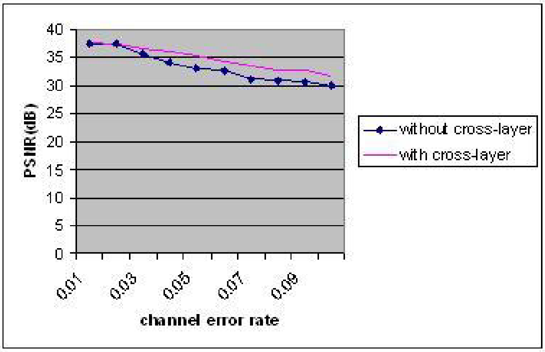

Fig. 5. Objective video quality of the News test video sequence

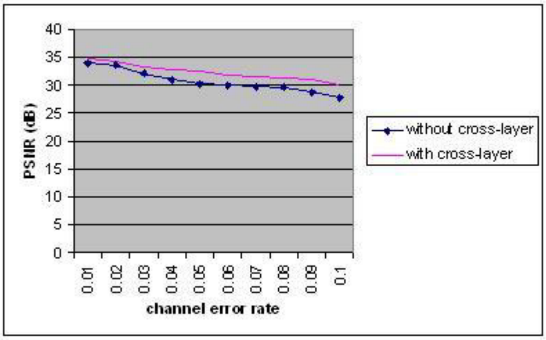

Fig. 6. Objective video quality of the Foreman test video sequence

parameters at the application layer (denoted by without cross-layer). In the entire three test video sequences, when channel state is good there is no tangible improvement in video quality, in case of proposed framework. When channel condition is moderate, the proposed cross-layer framework performs nearly 1 dB better in terms of PSNR than the case when no such proposed frame work is used. When channel condition is bad, the proposed cross-layer framework performs 1-2 dB better in terms of PSNR than the case when no such cross-layer approach is used. This improvement is more in case of medium and fast motion test video sequences than slow motion test video sequences. The improvement in the video quality in case of moderate and bad channel conditions is due to the variation of video rate at the application layer.

Fig. 7. Subjective quality test for three test video sequences

Subjective quality: We also evaluate the subjective quality of three test video sequences and results are as shown in Fig.4. The source format is cif (common intermediate format), where the pixel values are 352 × 288. The Fig.7(a), Fig.7(a') and Fig.7(a'') are the reference video frames, and are given here for the comparison purpose. Fig.7(b),

Fig.7(b’) and Fig.7(b”) are the screenshots of the three test video clips when they are reconstructed at the decoder with the channel error rate of 3 percentage, and in the absence of the proposed framework. The degradation in video quality is clearly visible at the central bottom part of Fig.7(b) , lower part of Fig.7(b’) and at lower part of Fig.7(b”). This shows that the viewers will experience a poor quality, while watching the video. When the proposed cross-layer based framework was used for video streaming, the results are as shown in the Fig.7(c), Fig.7(c’) and Fig.7(c”). This shows considerable improvement in the subjective quality of the video. The corresponding PNSR values and the gain in PSNR with the proposed frame work are listed in Table 1.

Table 1. PSNR values for the subjective quality test

	Mother – daughter	News	Foreman
Original (dB)	39.23	38.5	36.24
without cross-layer (dB)	36.64	33.30	30.30
with cross-layer (dB)	37.70	34.90	32.40
with versus without CL Gain (dB)	1.06	1.60	2.1

4 Conclusion and Future Work

In this paper, we have proposed a cross-layer framework for adaptive video streaming in wireless networks. The proposed cross-layer framework optimizes the transcoding rate at application layer depending upon the channel condition estimated using parameters associated with the data-link layer. We have evaluated the proposed cross-layer framework using NS-2 simulator with EvalVid framework for slow, medium and fast motion video sequences.When channel condition is moderate, the proposed cross-layer framework performs nearly 1 dB better in terms of PSNR than the case when no such proposed framework is used. When channel condition is bad, the proposed cross-layer framework performs 1-2 dB better in terms of PSNR than the case when no such cross-layer approach is used. Future work will be to implement the proposed framework in the real netwrok setup.

References

1. Shen, H., Sun, X., Wu, F.: Fast H.264/MPEG-4 AVC transcoding using power- spectrum based rate-distortion optimization. IEEE Transactions on Circuits and Systems for Video Technology 18(6), 746–755 (2008)
2. Bolot, J.-C., Turletti, T.: Experience with control mechanisms for packet video in the internet. Computer Communication Review (1998)
3. Ozcelebi, T., Civanlar, M.R., Tekalp, A.M.: Minimum delay content adaptive video streaming over variable bitrate channels with a novel stream switching solution. In: Proc. IEEE International Conference on Image Processing (ICIP 2005), vol. 1, pp. 209–212 (2005)
4. ITU-T Rec. H.274 and ISO/IEC 14497-10 (MPEG4-AVC), Advanced video coding for generic audiovisual services, ver. 1, May 2003; ver. 2, January 2004; ver. 3 (withFRExt), September 2004; ver. 4. July 2005

5. Jiang, M., Ling, N.: Low-delay rate control for real-time H.264/AVC video coding. IEEE Transactions on Multimedia 8(3), 467–477 (2006)
6. Li, Z.G., Pan, F., Lim, K.P., Feng, G.N., Lin, X., Rahardaj, S.: Adaptive basic unit layer rate control for JVT. In: Proceedings of the JVT-G012, 7th Meeting, Pattaya, Thailand (March 2003)
7. JM reference software, Maintained by K. Suhring (June 2008), `http://iphome.hhi.de/suehring/tml/`
8. Network Simulator, `http://www.isi.edu/nsnam/ns/`
9. EvalVid-A video quality evaluation tool-set, `http://www.tkn.tuberlin.de/research/evalvid/`