Asian Journal of Information Technology 15 (22): 4669-4677, 2016
ISSN: 1682-3915

QOS Aware Cross Layer Design for H.264-SVC Video Transmission Simulation over IEEE 802.11e Networks

[1]A. Srinivasan, [2]Dhananjay Kumar and [3]L. Arun Raj
[1]Department of Computer Science and Engineering, Thangavelu Engineering College, Chennai, India
[2]Department of Information Technology, Anna University, MIT Campus, Chennai, India
[3]Department of Computer Science and Engineering, B.S.Abdur Rahman University, Chennai, India

Abstract: The video streaming has become more popular in real-world multimedia communication. The demand for delivering a variable quality video over the wireless channels has become more challenging. These challenges motivate to introduce a scalability feature to enable video streams to adapt to fluctuations in the available bandwidths and to optimize the video quality. In this study, we simulate a Scalable Video Coding (SVC) with QoS aware cross layer design to improve the end to end delay, jitter and packet loss. We propose the cross layer design of the application layer and the MAC layer by tuning the MAC layer to the parameters from the application layer. Simulation results will show that the proposed system achieves significant gain in throughput and enhances the Quality of service.

Key word: Scalable Video Coding (SVC), H.264, cross layer design, communication, MAC layer

INTRODUCTION

The video streaming has become more popular with the advent of Internet in applications such as real-time video conferencing, web video streaming, mobile TV and video surveillance. These new applications have raised the demand for supporting various video devices with different capacities. Since the video streaming takes place in the Internet network space in wired and wireless environments, the network congestion and fluctuating bandwidth affect the smooth delivery of video across many devices with varying capacities. The demand is for best video quality whenever the user requires it and wherever it is required. The transmitted video must be able to cater to smart phone device or computer terminal or high definition TV workstation, each with different requirements. To overcome this impasse, Scalable Video Coding (SVC) technique is used. There are several standards proposed such as MPEG-2, H. 263 and MPEG for such applications to name a few. The H.264/AVC (Hsiao *et al*., 2012; Alabdul and Rikli, 2012 ; Khalek *et al*., 2011) a standard for video streaming offers temporal, spatial and quality scalabilities in single multi-layer stream. The scalable video coding (H.264/SVC) is an extension of advanced video coding. The objective of SVC is to encode a video stream with one or more subset bit streams, each can be decoded with a quality similar to the requirement of the device. SVC is flexible and adaptable because it only needs to encode a video once and the resulting bit stream can be decoded at multiple data rates and resolutions. The SVC provides three important features such as spatial scalability (adapt to spatial resolution), Temporal scalability (adapt to frame rate) and SNR (adapt to video quality) scalability.

Literature review

H.264/AVCAnd H.264/SVC overview: H.264/AVC standard partitions the picture into smaller units known as “macro blocks”. Each of these macro blocks covers an area of 16×16 pixels. Each macro block is sliced as P or B or I slice. These slices are coded in specific sequence for the receiver to decode them and display the moving picture. There is a flexible and adaptive Video Coding Layer (VCL) in H.264/AVC and Network Abstraction Layer (NAL). NAL is able to package VCL as NAL unit.

SVC is an extension of H.264/AVC. With SVC, it will be possible for various receivers to decode the video streams using different data rates and in to different resolution pictures. By this way SVC is adaptable and scalable. There will be only one data stream that can be decoded by different receivers as per individual requirements (Ke *et al*., 2008; Schwarz *et al*., 2007; Bianchi *et al* ., 2009 ; Schierl *et al*., 2007; Ke, 2012; Srinivasan *et al*., 2013).

IEEE 802.11e: IEEE 802.11e is a standard meant for many applications in wireless LANs especially addressing QoS

Corresponding Author: A. Srinivasan, Department of Computer Science and Engineering, Thangavelu Engineering College, Chennai, India

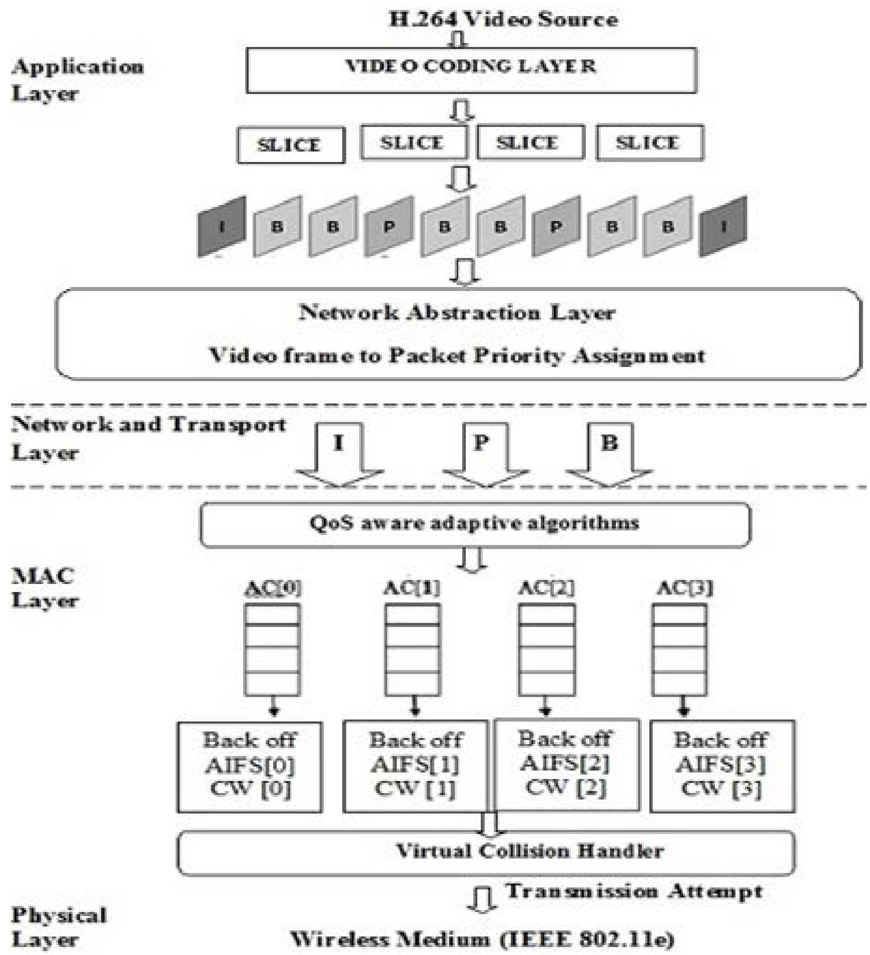

Fig. 1: Video mapped with different Access categories

issues. Simple applications such as web browsing, e-mail etc can tolerate some amount of delay in response time. The same may have adverse effects in video applications. 802.11e specifically addresses such problems in respect of multimedia, in particular video transmission. The 802.11e enhances DCF and PCF with the help of HCF Controlled Channel Access (HCCA) and Enhanced Distributed Channel Access (EDCA).

The 802.11e standard is with four different access categories namely, AC (0), AC (1), AC (2) and AC (3), each with different transmission priorities (Alabdul and Rikli, 2012). Individual ACs will be vying with each other to get access the channel for transmission as shown in Fig. 1. With the inherent inadequacies of the wireless channel, successful continuous multimedia transmission over wireless channel is a challenging task. IEEE 802.11e specifically overcomes these challenges by means of adjustments in MAC mechanisms such as Contention Window size, TXOP limit and data transmission rate. However, these adjustments notwithstanding, the video transmission over IEEE 802.11e requires modifications to sustain QoS.

The parameter set of EDCA consists of minimum and maximum Contention Window Size (CWmin, CWmax), Arbitration Inter Frame Size (AIFS) and Transmission Opportunity Limit (TXOP limit). The AC with the smallest AIFS has the highest priority and so it has higher probability of getting access to the medium. When a collision occurs among different ACs, the higher priority AC is granted the opportunity to transmit, while the lower priority AC suffers from a virtual collision (Eq. 1). AIFS is calculated as per the following equation:

$$\text{AIFS(AC)} = \text{SIFS} + \text{AIFS(AC)} * \text{Slot Time} \qquad (1)$$

Where SIFS is short inter frame space duration.

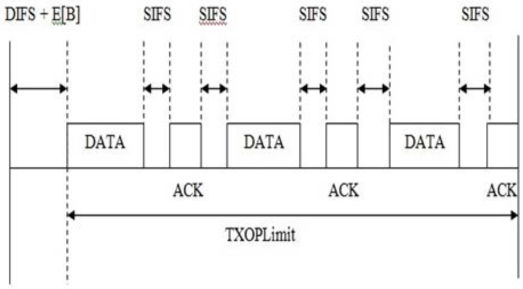

Fig. 2: Medium access when enabling TXOP limit

$$N_i = (P_i \times S_i) \quad (2)$$

The TXOP limit is computed based on Eq. 1 and 2:

$$T \times OP = N_i(\frac{M_i}{R_i}) + 2 * SIFS + ACK \quad (3)$$

Where Pi represents the mean data rate, Mi the packet size, Ri the channel data and SIFS is the service interval. The medium access is as shown in Fig. 2.

Thus, an Access Category (AC) with shorter contention period has more opportunity to occupy the channel. Although, EDCA provide QoS guarantees there is a challenge to determine how to configure these parameters to provide the specific quality of video. Even though there is a static method, it does not give absolute throughput and delay performance. Hence to provide best quality video, we need to tune EDCA based on the characteristics of the multimedia applications.

Cross-layer design: The layered system of Network protocols is primarily designed for wired networks. With the increased wireless networks traffic and with high-rate data dependent video streaming is becoming more of a norm in wireless networks (Alabdul and Rikli, 2012), it becomes necessary to have relook at the definition of layers and their functions. Strict layer based concepts are found to pose limitations on high data rate traffic. So a new approach at cross layer design for video streaming has been attempted by many (Khalek *et al*., 2011). These approaches sought to maintain functionalities associated with original layers allowing coordination, interaction and joint optimization of protocols across different layers. Traditionally while mobility management remained within a single layer there was a logical division of functions between network layer and link layer. With wireless networks, handling of

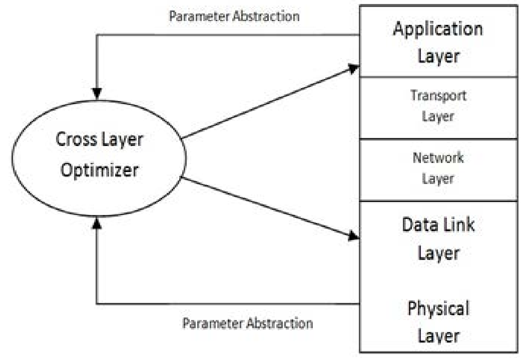

Fig. 3: Proposed cross layer architecture

mobility has become a challenge. Some layers appear well suited for handling mobility as compared to the others. However, mobility is found to be better managed across thelayers rather than in a single layer. So, there is a growing interest in cross layer design of wireless networks.

Proposed cross layer architecture: The proposed cross layer design for transmission of video is given in Fig. 3. As can be seen, in the proposed cross layer design, the application layer communicates with the MAC layer in data link layer. Before the actual transmission of video through wireless channel, the application layer sends prior information to data link layer. Due to this advance information, the data link layer prepares itself and keeps the algorithm ready for transmission of video. This MAC layer algorithm changes the parameters in the video content in such a way as to improve the quality of transmission. The parameters are data information indicating the characteristics of data flow and feedback information such as buffer states, channel conditions and acknowledgements provided by the user. The former is provided by the upper layers. This information is converged in MAC layer. The optimization in cross layer architecture is achieved as shown in Fig. 3.

The first step is to change Arbitrary Inter Frame Space Duration (AIFSD) that is the time gap between the idle state and the transmission state of the system. When this parameter is changed as per the priority of the data to be sent, the transmission becomes efficient. So automatically, other transmission parameters are improved.

Next to be changed is the T×OP. This controls the transmission opportunity of the data. T×OP is proposed to be kept constant in the proposed system.

Contention Window (CW), the third parameter is a part of AIFSD. Let us assume that there are two transmitters with the same value of CW. This will result in

Table 1: 802.11e EDCA parameter set

Priority	AC	Designation	AIFSN	CWMIN	CWMAX	TXOP limit
3	AC_VO	Voice	2	7	15	0.003008
2	AC_VI	Video	2	15	31	0.006016
1	AC_BE	Besteffort	3	31	1023	0
0	AC_BK	Background	7	31	1023	0

simultaneous transmission by both the transmitters resulting in collision. So, the receiver will not receive the data from any one of the transmitters. To keep a check on such occurrence, it is decided to keep the CW value adaptive. The various parameters are set as per Table 1.

There are two controls envisaged in this study, one as defined in Algorithm 1 and the other as defined in Algorithm 2 for spatial scalability and temporal scalability, respectively.

In control by algorithm 1, the system keeps comparing the terminal capacity and bandwidth. Based on the available conditions, the system keeps transmitting the video stream adaptively under Algorithm 1 control. If there is a report of loss of frames, the system switches to Algorithm 2 control and adaptively reduces the frame rate to suit the available bandwidth. The control through Algorithm 1 is by two steps. First to calculate the video resolution based on measured terminal capacity.

The next step is to change the frame rate based on available bandwidth. As specified in H.264, the priority of frames is I, P, B in that order. In Algorithm 1, the change levels of frame rate are 7.5, 15 and 30 fps. When the bandwidth is inadequate to support, Algorithm 2 takes over. It basically changes the frame rate. In this control, the system drops B frame first if the bandwidth is between total B and I frames.While doing so, the system calculates the number of B frames to be dropped.

If dropping of B frame is not adequate, the system drops two B frames with one P frame. If the bandwidth is found to be still inadequate, then Algorithm 2 takes over and calculates the bandwidth. At this time, system decides to transmit at another resolution and frame rate.

Algorithm1: VCL-calculate adequate badwidth:

```
Input
BWavail      Bandwidth Availabilty
Tn,Sm,Qo       Tn of Temporal resolution,Sm of spatial
Resoluations,Qo of Quality
 I  D n,m,o: Data rate of Temporal ,Spacial and
SNR
Begin Proceture
 T',S',Q' ←  Next level of T,S,Q
      Calculate teminal capacity to decide Tn,Sm,Qo
      If BWavaill > I  D  n,m,o then
        T',S',Q'  ß    S Tn,Sm,Qo
      Else
  T'ß-Algorithm 2
 End If
 End Procedure
```

Algorithm 2: NAL – Decide to drop frame

```
Input:
      Pf=P Frame Size
      Bf=B Frame Size
      N=Total number of P frames
M=Total number of B frames
      Begin Proceture
        E(t) _     Encoder Sending Rate
      BWavail _ Bandwidth Availabilty
      WhileE(t) >BWavaildo
      then
      Drop Bi Frame
      End If
      Else if Frame 'I' loss then
      GOTO Algorithm 1:
      Else
      Drop Bi and Pj=i+2Frames
      End While
      End Procedure
```

Algorithm 1 comes in to play during two conditions. When P and B frames are dropped due to low frame rate and when the bandwidth is adequate to support video transmission of higher quality indicated by no frame loss. The flowchart for Algorithm 1 and Algorithm 2 are shown in Fig. 4. Algorithm 3 calculates frame loss. Packet is identified by packet ID and packet IDs are helpful in calculating packet loss. Packet ID is also helpful in reordering of packets. In video transmission, it is imperative that we keep track of packets that are lost and also keep track of type of frame in the lost packets. Packet loss is estimated by the equation:

$$PL = \frac{\text{Number of packets}(t)\text{received}}{\text{Number of packets}(t)\text{sent}} \quad (4)$$

In the equation the variable t specifies the type of frame inside the packet. It is normally I or P or B type of frame. It is important to keep track of packet loss as this parameter is a major contributing factor for the quality of video. If more packets are lost, the rate of distortion of the video will be higher. A number of packets may be used for packeting data with respect to one frame. Frame loss is calculated as per the equation:

$$FL = \frac{\text{Number of frames}(I,B,P)\text{received}}{\text{Number of frames}(I,B,P)\text{sent}} \quad (5)$$

Algorithm 3: Frame Loss Estimation

```
Input:Number of frames (I,B,P)Sent.
Output: Number of frames (I,B,P) received.
Begin Proceture
      MAX = Maximum play-out buffer size
      new_arrival_time (0) := orig_arrival_time(0)
      ForEach Frame m
      If (m is lost)
      new_arrival_time(m) := new_arrival_time(m-1) + MAX
      End If
      Else If (interframe_time(m) >MAX)
      Frame mis marked lost;
```

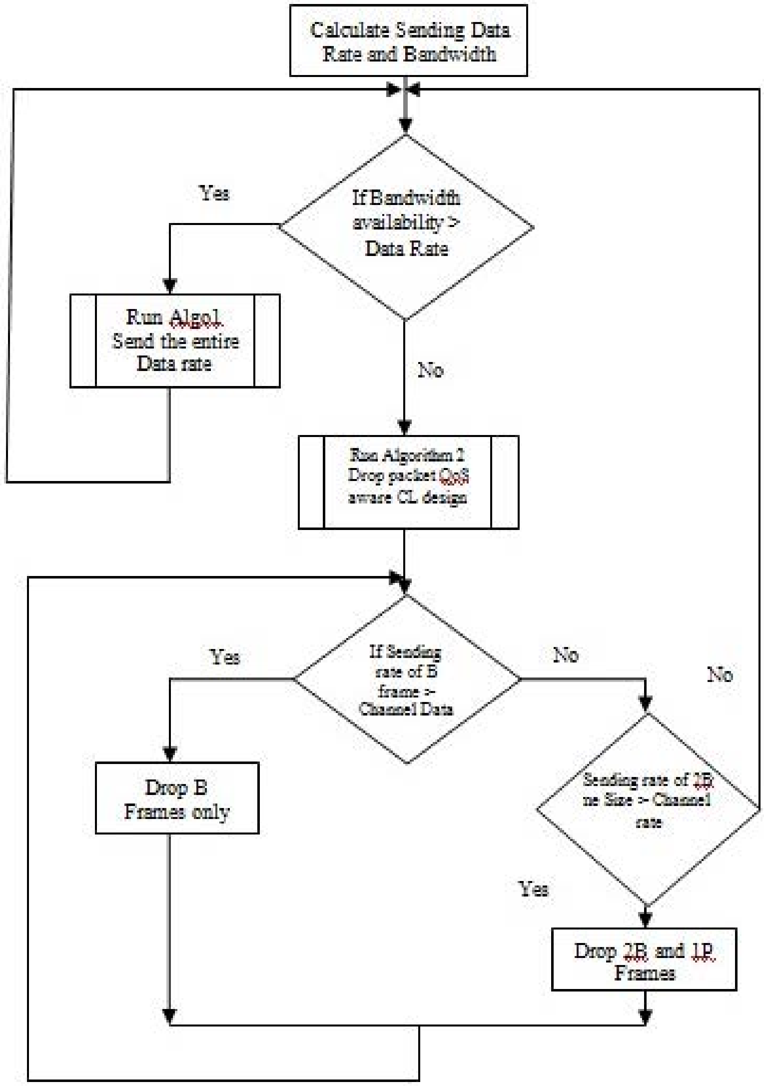

Fig. 4: Flowchart of Algorithm 1 and 2

```
    new_arrival_time(m) := new_arrival_time(m-1)+MAX
    End If
    Else
    new_arrival_time(m) := new_arrival_time(m-1) +
(orig_arrival_time(m) - orig_arrival_tm(m-1))
    End If
    End For
End Procedure
```

These two equations 4 and 5 are used to estimate the packet and frame loss in algorithm 3.

RESULTS AND DISCUSSION

Simulation and performance analysis: The simulation is carried out for evaluation of SVC traffic over 802.11a, 802.11e and also 802.11e with our algorithms. This research is implemented using Network Simulator 2 (NS2), Joint Scalable Video Model (JSVM) and Scalable Video-streaming Evaluation Framework (SVEF). NS2 is an open source, object oriented discrete event simulator. It is written in C++ having Object Tool Command Language (OTcl) for command and configuration interface. JVSM is also open source software written in C++. JVSM is considered to be reference software for Scalable Video Coding (SVC) project of Joint Video Team (JVT) of the ISO/IEC Moving Pictures Experts Group (MPEG) and the ITU-T Video Coding Experts Group (VCEG). SVEF is mixed online/offline open source software that is mainly used to evaluate the H.264/SVC video streaming.

It is written in C and Python [2,4,5,10,14,25]. The test video source used is Foreman. It is in YUV CIF(352-288) format. It has 300 frames. The test video source is

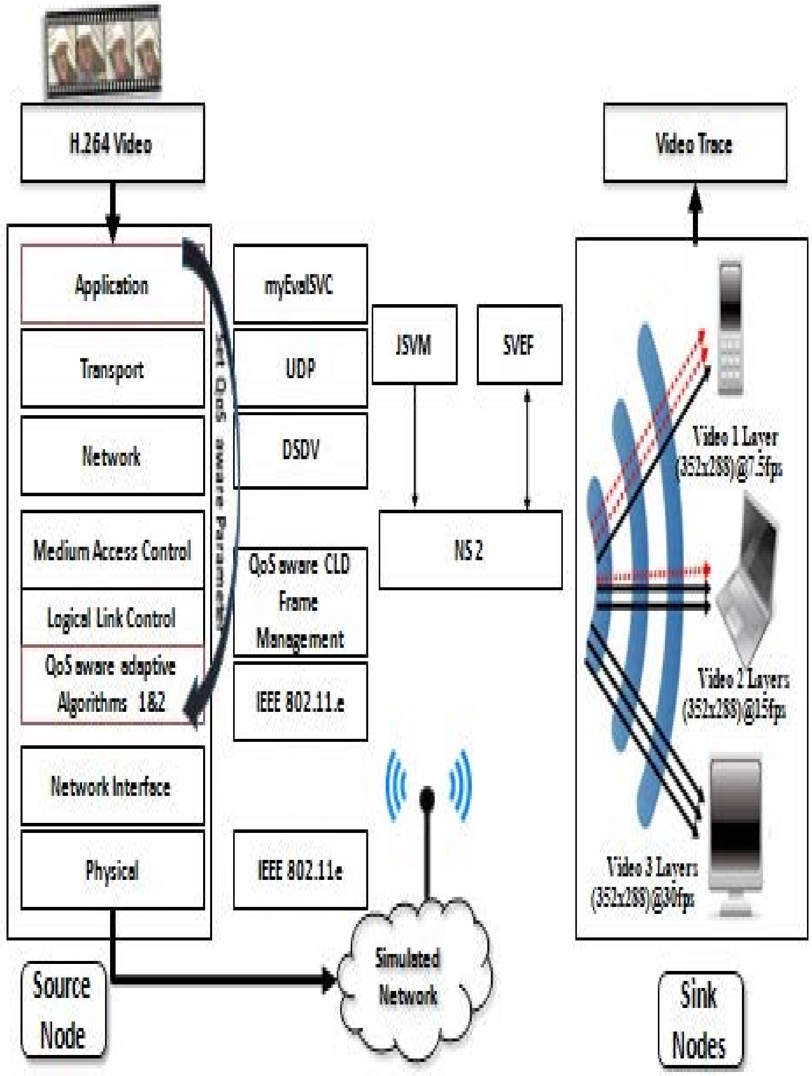

Fig. 5: Framework for simulation

Table 2: Video parameters for Foreman video

Layer	Resolution	Frame rate (per second)	Bit rate (kbps)	Dependency id(Did),Temporal Id(TId), Quality id(QId)
0	352×288	7.5	514.10	(0,0,0)
1	352×288	15.0	548.70	(0,1,0)
2	352×288	30.0	588.20	(0,2,0)

encoded with JSVM (Version 9.19) having only temporal scalability enabled. The video parameters are listed in Table 2.

In the simulation scenario, many factors such as number of nodes, movement model, number of senders, video trace, queue length and distance between the source and sink nodes are taken into account. Such clear description of scenario helped us to have a clear and comprehensive insight into many factors of network functions such as the way the network behaves and the effect of various parameters and conditions on the network performance.

The types of scenarios used in simulation are with static and dynamic topologies. These are implemented with parameters as listed in Table 3. We have used JVSM and SVEF in preparation of video traces. Multiple video traces are used in network scenarios. The graces used are generated using encoded video with minimum grain scalability (3, 9, 11, 17-19) (Klaue *et al*., 2003). The simulation framework is shown in Fig. 5-7.

It can be seen from the Table 4 that the proposed system gives better results for the frame loss. That is as compared to 110 frames lost in 802.11a and 70 frames lost

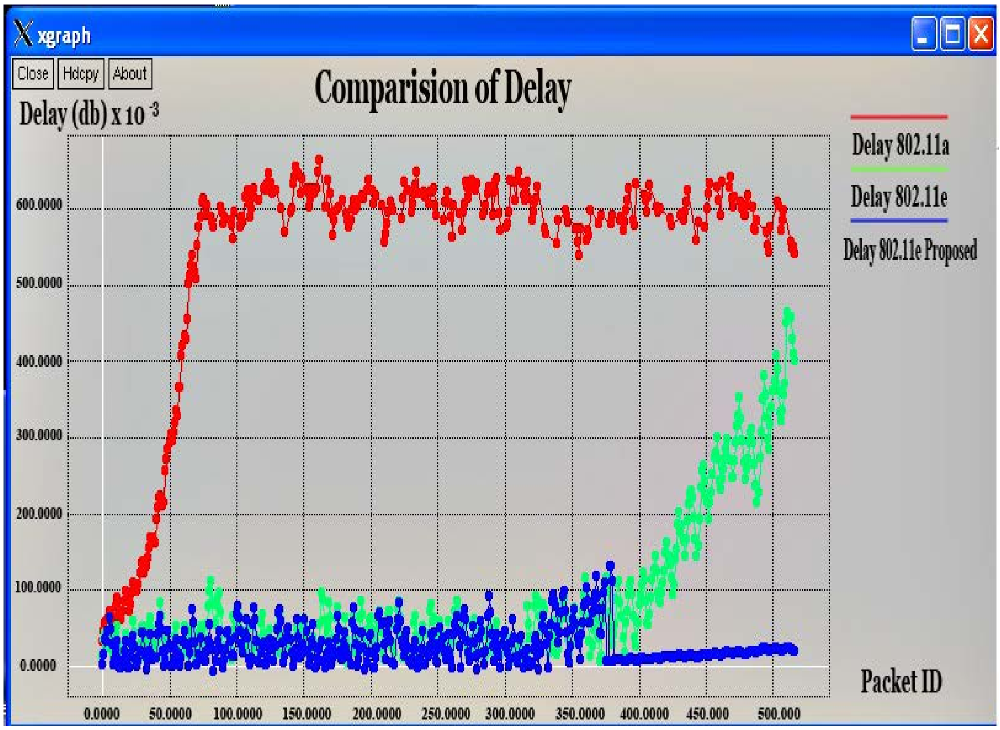

Fig. 6: Comparisons of delay

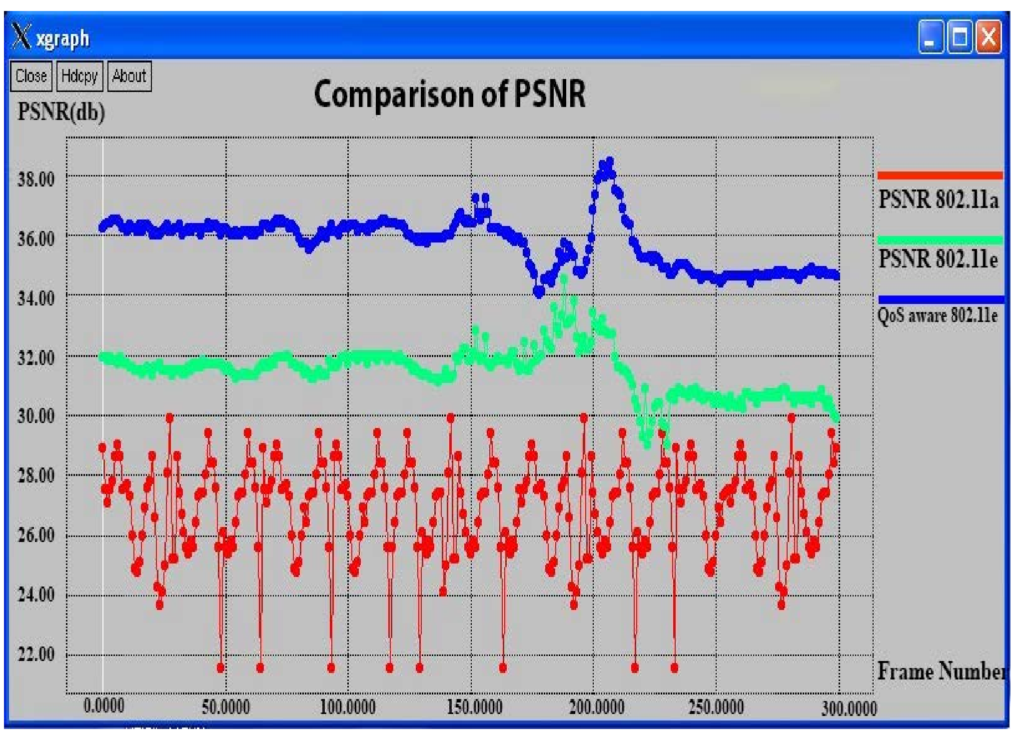

Fig. 7: Comparisons of PSNR

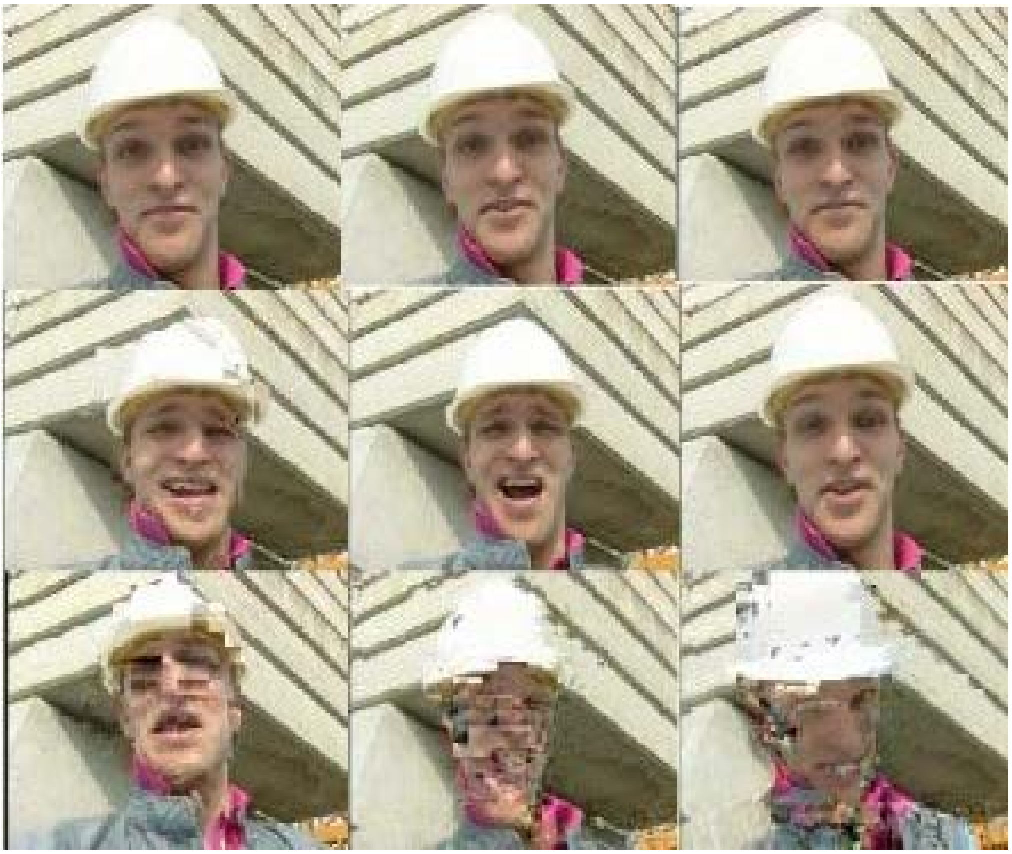

Fig. 8: Visual Comparison of the schemes: Proposed QoS aware IEEE 802.11e (Top),IEEE 802.11e (Centre) and IEEE 802.11a (Bottom)

Table 3: Parameters used in simulation

Number of frames	300
Video file	Foreman.yuv
Routing protocol	Destination Sequenced Distance Vector (DSDV)
Video resolution	352×288 (CIF)
Data rate	1 Mbps
Play-out delay	5 sec
Sending rate	0.2 Mbps
Transmission power	15 db
Channel type	Wireless channel
Queue size	50
Network interface type	802.11a, 802.11e
MAC type	802.11a, 802.11e
Interface queue type	PriQueue/QoS aware priqueue
Antenna model	Omni antenna

Table 4: Comparison of proposed IEEE802.11a, IEEE802.11e and QoS aware IEEE802.11e

	Average PSNR(dB)			No. of frame lost			
WLAN	Y	U	V	I	P	B	Total
IEEE 802.11a	27.1686	31.8856	36.5632	25	15	70	110
IEEE 802.11e	31.0200	37.8153	39.9236	5	10	55	70
QoS aware proposed 802.11e	35.2521	38.7864	40.6786	0	6	40	46

Table 5: PSNR, delay, Jitter, frame loss and packet loss trace values using NS-2 simulator

Parameters	IEEE 802.11a	IEEE 802.11e	QoS aware 802.11e
Average PSNR (dB)	27.23	31.13	35.25
Endtoend Delay(s)	0.6214	0.3521	0.06192
Cumulative Jitter(s)	0.8281	0.0567	0.03024
Frame loss in numbers	110	70	46
Packet loss rate	51.4%	14.23%	8.23%

in 802.11e, the proposed system has the frame loss of only 46 frames. It has to be further observed that there is totally no I frame loss in the proposed system. The loss of I frame in a video has to be avoided as it may result in lesser video quality.

It can also be seen that the PSNR is consistently higher than 35 dB in the proposed system ensuring excellent video quality. It can be seen from the Fig. 8 that the PSNR is higher in the proposed system as compared to the 802.11a and 802.11e systems. It can be seen from Table 5 that the proposed system reduces the delay drastically compared to the conventional systems as shown in Fig. 6 and 7.

CONCLUSION

A cross layer design for implementing video streaming with better quality has been proposed and simulated in NS-2 simulator. The numerical results observed during simulation validate the usefulness of the proposed work. The QoS aware cross layer design was targeted to achieve acceptable PSNR while meeting constraints of wireless network. One of the main features of the design is to ensure that base frames have a greatest priority whenever the network experiences low bandwidth, reducing the total number of dropped base and enhancement frames, reducing the average delay as much as possible and maintaining jitter within acceptable limits. It is found that the PSNR has been improved by 15% with our design while the delay is reduced by 10%. As an enhancement, the work can be extended to transmit real time medical videos in 4G.

REFERENCES

Alabdul, K.M.N. and N.E. Rikli, 2012. QoS provisioning for H. 264/SVC streams over Ad-Hoc Zig Bee Networks using cross-layer design. Proceedings of the 8th International Conference on Wireless Communications Networking and Mobile Computing (WiCOM), September 21-23, 2012, IEEE, New York, USA., ISBN: 978-1-61284-684-2, pp: 1-8.

Bianchi, G., A. Detti, P. Loreti, C. Pisa and F.S. Proto et al., 2009. Application-aware H. 264 scalable video coding delivery over wireless LAN: experimental assessment. Proceedings of the 2nd International Workshop on Cross Layer Design, 2009 IWCLD.'09, June 11-12, 2009, IEEE, New York, USA., ISBN:978-1-4244-3301-8, pp: 1-6.

Hsiao, Y.M., C.H. Chen, J.F. Lee and Y.S. Chu, 2012. Designing and implementing a scalable video-streaming system using an adaptive control scheme. IEEE. Trans. Consum. Electron., 58: 1314-1322.

Ke, C.H., 2012. My Eval SVC: An integrated simulation framework for evaluation of H. 264-SVC transmission. Trans. Internet Inf. Syst. (TIIS.), 6: 379-394.

Ke, C.H., C.K. Shieh, W.S. Hwang and A. Ziviani, 2008. An evaluation framework for more realistic simulations of MPEG video transmission. J. Inf. Sci. Eng., 24: 425-440.

Khalek, A.A., C. Caramanis and R.W. Heath, Jr., 2011. Joint source-channel adaptation for perceptually optimized scalable video transmission. Proceedings of the 2011 IEEE Conference on Global Telecommunications (GLOBECOM. 2011), December 5-9, 2011, pp: 1 5-10.1109/GLOCOM.2011.6134145.

Klaue, J., B. Rathke and A. Wolisz, 2003. Evalvid-A Framework for Video Transmission and Quality Evaluation. In: Computer Performance Evaluation. Modelling Techniques and Tools. Peter, K. and H.S. William (Eds.). Springer, Berlin, Germany, ISBN:978-3-540-40814-7, pp: 255-272.

Schierl, T., T. Stockhammer and T. Wiegand, 2007. Mobile video transmission using scalable video coding. IEEE Trans. Circuits Syst. Video Technol., 17: 1204-1217.

Schwarz, H., D. Marpe and T. Wiegand, 2007. Overview of the scalable video coding extension of the H. 264/AVC standard. IEEE. Trans. Circuits Syst. Video Technol., 17: 1103-1120.

Srinivasan, A., L.A. Raj and D. Kumar, 2013. Cross layer implementation with multiple descriptive coding to support video streaming in IEEE 802.11e networks. Proceedings of the IEEE International Conference on Recent Trends in Information Technology, July 25-27, 2013, IEEE, New York, USA., ISBN:978-1-4799-1024-3, pp: 434-440.