

ENHANCED NETWORK THROUGHPUT OF WIRELESS NETWORK FOR DIGITAL VIDEO TRANSMISSION

S. M. Koli¹, R. G. Purandare², S. P. Kshirsagar³ and V. V. Gohokar⁴

¹Department of Electronics and Telecommunication Engineering,
Smt. Kashibai Navale College of Engineering, Pune, India.

sanjaykoli@yahoo.com

²Department of Electronics and Telecommunication Engineering,
Vishwakarma Institute of Information Technology, Pune, India.

purandare.radhika@viit.ac.in

³AnchorTek Techno Consultancy Pvt. Ltd. Pune, India.

shirish@anchorteksys.com

⁴Department of Electronics and Telecommunication Engineering,
Shree Sant Gajanan Maharaj College of Engineering, Shegaon, India.

vvgohokar@ssgmce.ac.in

ABSTRACT

Digital video transmission using wireless technology is an important functionality in multimedia communication. Such wireless transmission considers various parameters of channel like reliability, quality of service (QoS), transmission rate, transmission delay etc. User datagram protocol (UDP) is a fast transmission protocol used by most of the real-time applications as it is suitable for delay sensitive applications like video and audio transmission. UDP does not provide flow control or error recovery and does not require connection management. Since more applications with real-time constraints are introduced day by day, both over wired and over wireless, some improvements should be made in order to obtain better end to end performance. This paper surveys the literature on real time communication over wireless and suggests an intermediate solution providing more reliability in transmission using adaptive error control mechanism. The idea used is to inject an adaptive amount of redundant packets (RPs) in every sent block (or datagram) in order to achieve a desired recovery rate at the receiver without using any retransmission mechanism. The technique used is called adaptive forward error control (AFEC) and is implemented at physical layer. The network simulator NS-2 is used to evaluate AFEC and the simulation results indicate that the suggested methods can guarantee enhanced network throughput for video transmission in the high packet loss wireless network.

KEYWORDS

Network Protocols, Digital Video Transmission, Wireless Network, End-to-End Delay, AFEC, PDR, NS-2

1. INTRODUCTION

In the last decade, there has also been a rapid development in the digital video communication field. Combination of advances in digital video compression and digital wireless communications resulted in a new service area called video over wireless. Due to the proliferation of multimedia on WWW and broadband wireless network, wireless video communication has received great interests from both industry and consumers for application such as

- Video conference;
- Power Point and other visual displays to be shared.

- Video-on-demand or multimedia streaming services
- To reach anyone immediately.

These multimedia services are now available on the smart handheld devices of user and demand the wireless link to be robust to error [1]. For the real time multimedia signal transmission in wireless medium, available radio resources are limited and hence there is a need of development of adaptable radio interfaces to adapt the variations in the wireless link, in order to optimize network performance [2, 3] which will result in good end-to-end wireless network performance.

The end-to-end flow and congestion control mechanisms are required to analyze the end-to-end performance of a protocol that has been designed and optimized for wired and wireless networks [3, 8, 25]. The deleterious effects of noise, interference, jamming, fading, and other channel impairments [1, 7] can be effectively reduced by using channel coding and interleaving techniques [24]. Section 2 gives a brief overview on channel coding. Various wireless transmission techniques related to the Internet transport layer protocols available in the literature are evaluated in section 3. Section 4 describes implementation of different modules proposed. The article highlights the results with discussions in section 5 and ends with conclusions in section 6.

2. CHANNEL CODING

The Shannon's model of digital communication system employing FEC scheme is shown in Figure 1 [7][9][21].

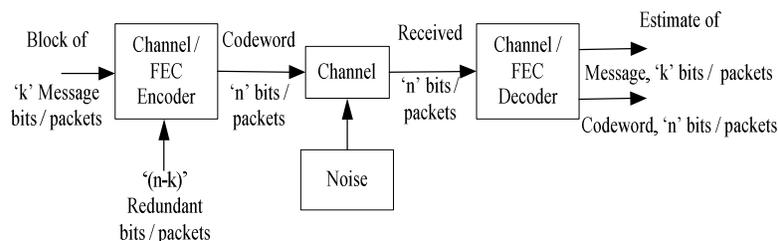


Fig. 1 Shannon's model of communication [7]

The idea of channel coding is to introduce some controlled redundancy into the original data to allow reconstruction of damaged blocks at the receiver. The redundant data is generated from the original data using techniques from coding theory, where two such techniques are well known in the literature as feed-forward error correction (FEC) and automatic repeat request (ARQ) [4, 5, 7, 10].

2.1 FEC

The channel encoder in the transmitter employing FEC, accepts block of 'k' message bits or packets and adds redundancy R according to the prescribed rule as $R = (n-k)$ in the controlled manner. The redundant data can be used to recover lost source data at the receivers. A receiver can reconstruct the original source data once it receives a sufficient number of packets. The combined goal of the channel encoder and decoder is to minimize the effect of channel noise. No communication is done by the receiver with transmitter after decoding, hence FEC require simplex link. FEC schemes are further classified as media independent FEC and congestion control [2, 7, 14, 17].

2.2 ARQ [4, 5]

Upon the detection of an error in the transmitted packet, the receiver requests the sender to repeat transmission of the lost packets (LP), which necessitates the use of the return channel and results in a great network load and hence are typically not acceptable for live audio and video applications and out of the scope of this paper. In next section 3, various Internet transport layer protocols are revisited.

3. INTERNET TRANSPORT LAYER PROTOCOLS

The transport layer [4-6] provides end-to-end segment transportation of messages and is reassembled back into the original message at the destination nodes. Examples of transport protocols are transmission control protocol (TCP), the user datagram protocol (UDP). TCP/IP and UDP/IP are the core of today's Internet transport layer [3, 5, 6, 20].

3.1 TCP

TCP is the reliable transport layer protocol and most Internet applications today rely on the TCP. The TCP/IP is the most popular protocol suite where stream data transfer reliability, efficient flow control, full-duplex operation and multiplexing is required, but it is not suitable for real time transmission.

3.2 UDP

UDP is connectionless transport layer protocol which doesn't require connection establishment prior to data transfer and runs on top of IP networks. UDP data units are called data-grams; also referred to as blocks. The applications where reliability is not critical or with strict transmission delays constraints such as real-time traffic are carried by the UDP/IP protocol. Here the datagram is send with the hope that receiver will be able to handle it. Hence, the UDP protocol is an unreliable protocol and it is suitable for broadcast of data [4, 7, 10, 13]. But UDP has a disadvantage of losing many packets. But it is not critical for real time applications. It does not provide sequence number management while exchanging data-grams and does not guarantee orderly transmission. It also does not offer capabilities for congestion or flow control [6]. The protocol in charge of providing sequence number control is the RTP protocol running on top of UDP (RTP-on-UDP). UDP contains no ACK mechanism; therefore, the lost data-grams can be recovered only by lower or upper layers, including the application layer [20]. Each UDP datagram is composed of a header and a payload (user data). In the payload the data coming from the layer above is encapsulated. UDP takes messages from the application process, then adds the source and destination port number fields, the length field and finally the checksum. The resulting segment is passed to MAC layer where it is encapsulated into an IP datagram.

When the datagram is received at the receiver it is divided into an IP header and IP payload. The latter is passed to the transport layer i.e. the UDP layer. UDP then uses the port number contained in its header to deliver the data to the correct application. Therefore, if a datagram does not arrive to the receiver, there is no possibility to recover it or to ask for a retransmission, because the receiver does not have any information about the sent packets. It has only the source port which takes into account the equivalent field at the IP header to protect the header.

3.3 RTP-on-UDP [20]

RTP is a generic transport protocol which is independent of applications. But is implemented at the application layer and is designed to handle real-time traffic on the Internet. RTP does not have a delivery mechanism; it must be used with UDP. RTP stands between UDP and the application program. The main contributions of RTP are time-stamping, sequencing and mixing facilities. Figure 2 shows how the data from the application is encapsulated as it is lowering in the UDP/IP protocol stack.

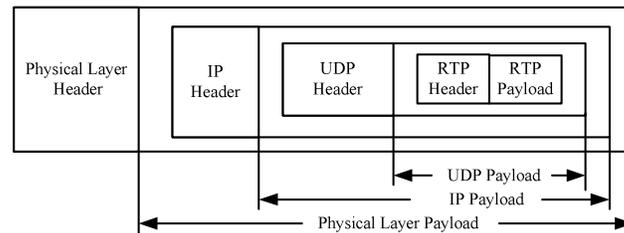


Fig. 2 Protocol Encapsulation

Thus, the RTP payload is filled with data coming from the application layer. Then this RTP packet plus the RTP header are passed to the lower layer, the UDP layer. The UDP layer takes this RTP packet and adds the UDP header. Then this UDP datagram is passed to the IP layer who adds also its header. Finally the IP datagram is passed to the physical (PHY) layer which also adds its header. Applications typically run on RTP-on-UDP to make use of its multiplexing and checksum services. Therefore, RTP protocol provides end-to-end delivery services for data with real-time characteristics, such as audio and video [8]. But, note that RTP itself does not provide any mechanism to ensure timely delivery or provide other quality-of-service guarantees, consequently RTP have to rely on lower-layer services to do it. A RTP packet consists of a fixed RTP header and a RTP payload.

The receiver has, thanks to RTP, enough information to manage the packets. But suppose that a packet have been lost somewhere in the network due to congestion or to link, then once the sender notice that a packet has been lost, it has to handle it somehow. Some intermediate solution providing more reliability to the communication between applications running on top of UDP and at the same time support its fast connection quality using already existing protocols is required. FEC coding commonly serves the purpose and included in the PHY layer design of wireless links [13]. This error control mechanism attempts to minimize the visual impact of LPs at the destinations [1, 3, 6].

FEC can be applied both at the bit-level and on the packet-level. Normally packet-level FEC is applied to end-to-end communication whereas bit-level FEC is used on a specific link. Packet-level FEC [3, 4, 9, 10, 16, 24] is chosen in this paper because more information about the requirements of the application can be used. Suppose that an amount of k data packets has to be sent in a block. To this data, $R = (n - k)$ (RPs) or parity check packets are appended. The RPs are calculated using block coding theory. Therefore, finally a total of $n = k + R$ packets are sent per block.

Recovering ' k ' packets at the receiver is achieved if no more than ' R ' packets are dropped. If more RPs are added than needed, then the network load will be unnecessarily increased. Nevertheless, if less RPs than needed are added then the LPs will not be recovered and also the network will be loaded with redundancy traffic. Therefore, there is a clear compromise between the obtained throughput and the network load. The relation between the packet delay and throughput with the network load can be understood with the help of Figure 1 below [5].

From the network theory, it is clear that when the load is much less than the capacity of network, the delay is at a minimum, whereas when the load reaches the network capacity the delay increases sharply. The delay becomes infinity when the load is greater than the capacity. Delay has a negative effect on the load and consequently the congestion. The wireless network throughput is the average rate of successful bytes delivery over a channel. Further, the throughput almost linearly increases if the network load doesn't exceed the network capacity resulting into no congestion. But the throughput decreases logarithmically reaching almost to zero if the network load exceeds the network capacity resulting into congestion. From the point of view of the packet loss probability (PLP), the added redundancy makes it to decrease. The amount of RPs has to be chosen trying to get as small packet loss rate (PLR) as possible. Normally RS codes are used, but another such as parity or Hamming codes could be also used

[15]. FEC is independent of the nature of the application data, but is implemented on the application layer. The FEC packet is obtained by placing the FEC header and the FEC payload in the RTP payload [5], as is shown in Figure 3. The FEC payload is composed by the application data and the FEC header is constructed by placing on it the redundant packets and is supposed to be sent to the lower layer at the UDP/IP stack, the RTP protocol.

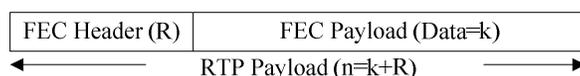


Fig. 3 Encapsulation of FEC-on-RTP protocol

All the concepts considered so far can be represented by block schematic in relation with the TCP/IP network model of computer communication network as shown in Figure 4.

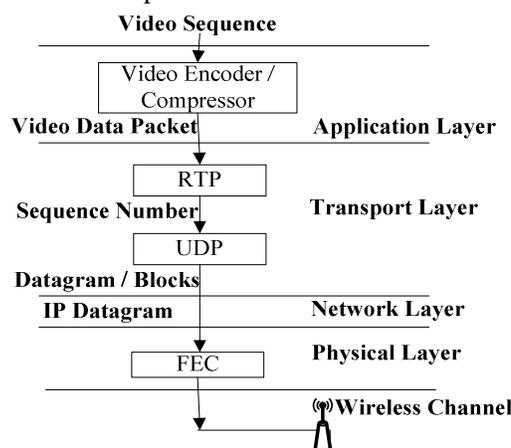


Fig. 4 Video over Wireless channel in relation with network model

3.4 Adaptive forward error correction (AFEC)

Good end-to-end wireless network performance will not be possible without a truly optimized, integrated and adaptive network design [1, 3]. Hence, instead of fixing a level of overhead that can cope with worst-case conditions, adaptive error control mechanisms let the error protection vary as the conditions vary i.e. the overhead is always adapted to the current conditions of channel. As has been mentioned, the UDP receiver does not have any information about the sent packets. RTP running on top of UDP provides sequence number value to each sent packet. Once the receiver has the sequence numbers it can know whether a packet has been lost or not. But, if there is no retransmission mechanism in UDP, how could this LPs be recovered?

In particular, in this paper, the number of RPs will be varied in response to the changing PLP trying to keep a specific PLP quality after decoding. Thus, an adaptive feature is added to FEC and then it is renamed as AFEC. The idea of AFEC is to inject an amount of redundant packets in every sent block (or datagram) in order to achieve a desired recovery rate at the receiver without using any retransmission (of data) mechanism. The specific amount of redundancy is updated by the sender based on channel loss probability measurements as shown in Figure 5, which explains the method to implement proposed AFEC. The block probability calculator along with control modules is called as intelligent control module (ICM). FEC along with this ICM is called as AFEC. This complete wireless system is simulated using NS-2 [23].

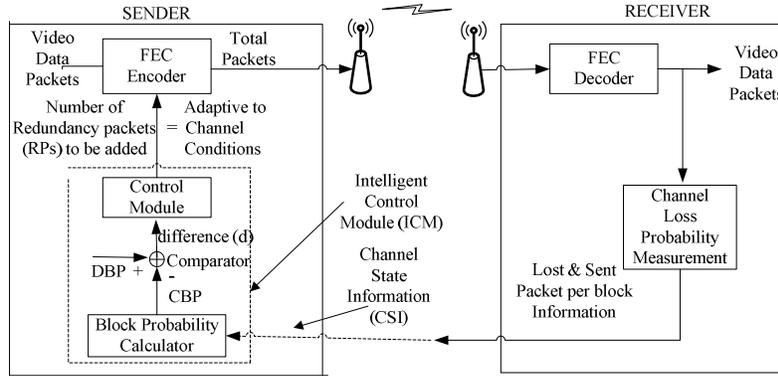


Fig. 5 Concept of Adaptive FEC and Its Implementation

In the simulation, the behaviour of ICM is checked for the number of LPs per block is constant, for adding loss probability in a link, further increasing loss probability to understand the system behaviour and adding competitive sources to cause congestion. All these four simulation cases are tested for the three different ICMs proposed in next section. The information regarding number of lost packets due to error is referred as channel loss probability measurement and indicates lost as well as send packets per block information and is also called as channel state information (CSI). These measurements are made by the receiver when a new block is received. The receiver sends this information as CSI to the sender who is the one in charge of updating the necessary amount of redundancy. Thus, if a block turns out to be corrupt by the loss of some of its packets, the receiver will be capable of recovering the lost information, if no more than the amount of introduced RPs has been lost or corrupted [3,14]. Thus, the RTP has been chosen as upper level protocol to provide flow control in AFEC technique to provide error management, which answers the question raised above. To obtain the appropriate number of RPs from the calculated block probability (CBP) various ICMs or feedback control systems can be used [16,17]. In this paper ICMs such as a Delta intelligent control module (DICM), a Adaptive Delta intelligent control module (ADICM) and a Adaptive Delta Sigma intelligent control module (ADSCM) are proposed, details of which is explained in section 4.

Instead of AFEC/UDP other solutions have been proposed in the literature as reliable UDP (RUDP) or UDP Lite [18,19] but they are out of the scope of this paper. RUDP [18] is suitable for transport telecommunication signalling and is layered on the UDP/IP. UDP Lite was proposed to prevent packets loss at the receiver side if channel errors are only located on the packet payload. Therefore, errors detected in the packet header result in a discarded block whereas errors detected in the packet payload does not result in a discarded block [19]. Enhanced AFEC (EnAFEC) [22] is also proposed but it estimates the suitable smoothing factor value to determine the average queue length according to the packet loss rate over the burst wireless error network.

4. IMPLEMENTATION OF ICM

The CSI from the receiver is used by the sender to calculate the optimal number of RPs to be added to FEC code. Feedback information is updated only when a block has been totally received. The sub-block of sender in Figure 5, called ICM does following calculations.

- CBP calculation. It is calculated by the block probability calculator using Equation 1 [12].

$$P_{(CBP)}(n, k) = \sum_{i=0}^{k-1} \binom{n}{i} (1-P)^i (P)^{n-i} \left(\frac{n-i}{n}\right) \quad (1)$$

Where :

'P' is the estimated packet loss probability without FEC, 'n' is the total number of packets per block, 'k' is the data packets per block and $\frac{n-i}{n}$, which gives the average number of lost packets when the losses cannot be recovered. Therefore, with this factor the equation 1 gives the average packet loss probability and without this factor the formula gives the block loss probability.

- CSI calculation is done with the help of LP per block and sent packet per block information estimated by the channel loss probability measurement unit at the receiver. With respect to equation 1, CSI can be mathematically represented as, $\hat{P}_{CSI}(n, k)$.
- DBP calculation. It is the desired block probability and is set to 0.005 because it is desired to achieve a small block loss probability in the simulations. With respect to equation 1, DBP can be mathematically represented as, $\hat{P}_{DBP}(n, k)$.
- Error $d = DBP - CBP$ and is calculated by the comparator. This error could be positive or negative based on CBP and is represented by $\pm \delta$ (Delta). This error can be mathematically represented as, $d(n, k)$.
- RPs calculation. The error 'd' is applied to the control module which calculates number of RPs to be added by FEC encoder and acts as adaptive parameter for AFEC. These RPs (R) are inserted into the data block 'k' resulting in a total of 'n' packets to be sent to the network.

4.1 Delta intelligent control module (DICM)

The DICM is the simplest control module and uses only the information about the number of LPs and RPs per block. The operation of this system is based on a step by step increments or decrements of the ICM output i.e. number of header packets (HPs), which depends on the error 'd' which is either $+\delta$ or $-\delta$. The number of header RPs in-turn will be incremented or decremented by one. Hence the name to the scheme is given as Delta module and is shown in Figure 6.

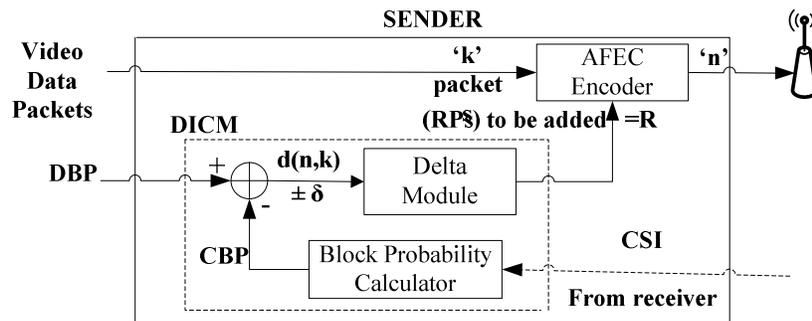


Fig. 6 Delta intelligent control module (DICM)

Suppose that the number of LPs in the received block is denoted by the 'LP' variable and the number of HPs for the same block is denoted by 'HP'. Then working of DICM can be understood with the help of the flowchart as shown in Figure 7. For the case $LP > HP$, the header is increased by one packet because it is quite probable that in the next block a similar amount of packets will be lost as uniform distribution of the network errors is used. As increment or decrement of the number of HPs is made one by one based on $+\delta$ or $-\delta$, this method is not the most optimal method.

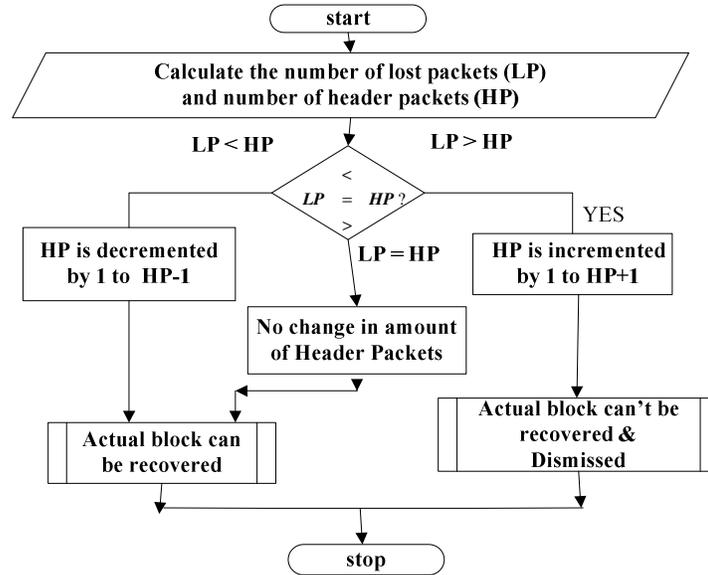


Fig. 7 Flowchart showing working of DICM

4.2 Adaptive Delta intelligent control module (ADICM)

To make DICM optimal, increment or decrement in the number of RPs should be done in steps of size bigger than one packet in order to follow better the changes in the number of LPs per block. Additional HP size controller is added to the ICM of DICM which will increase or decrease the number of HPs by more than one, based on the error 'd' (i.e. $+\delta$ or $-\delta$) which in turn controls the increment or decrement of HPs by more than one. This information is used to get the number of RPs to be added. Hence the technique is named as Adaptive DICM (ADICM). The complete schematic is as shown in Figure 8.

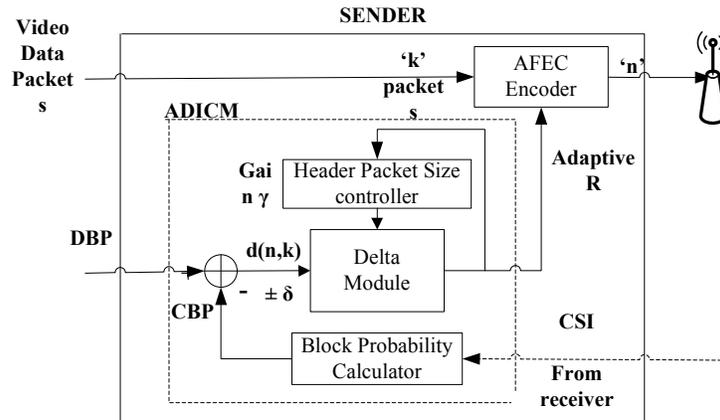


Fig. 8 Adaptive Delta intelligent control module (ADICM)

The RPs (R) are introduced by the sender in every block in order to achieve the small number of dismissed packets in the receiver side after decoding as possible. In the proposed ADICM the relation between the output and the input of the system is given by a constant, called ' γ ' and the system has to be tuned by fixing this gain γ based on the results of the simulations. The results of several simulations using NS-2 for the two main cases mentioned before are:

- For the first case, where the number of LPs per block is constant, the value of γ is increased starting from 0 until the header value exceeds the constant number of LPs per block and remains stable. The obtained value is $\gamma = 170$.
- For the case of variable number of LPs in a block the scenario chosen to tune the ICM is a lossy link with uniform probability distribution and loss probability equal to 0.03.

4.3 Adaptive Delta Sigma intelligent control module (ADSICM)

The ADSICM introduces a integrator (a low band filter) in the ADICM system in order to hide the effect of the residual noise and follow better the changes in the number of LPs per block. The block diagram of such a new system is shown in Figure 9.

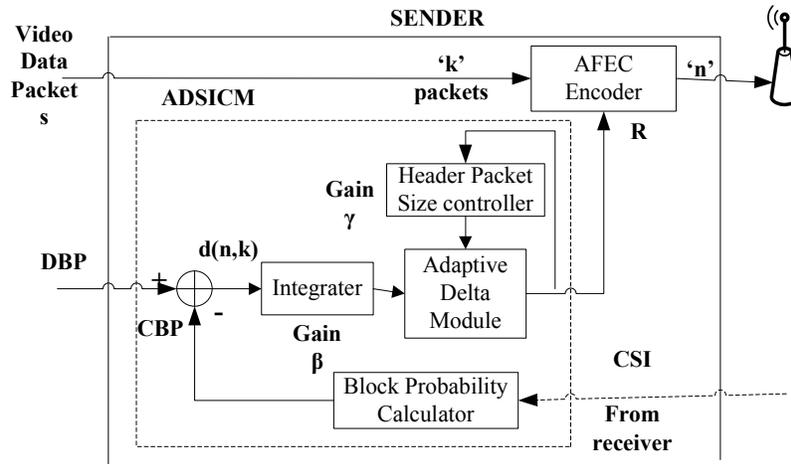


Fig. 9 Adaptive Delta Sigma intelligent control module (ADSICM)

In this case, both the integral gain β and the adaptive gain γ , has to be fixed. Also for this ADSICM system the choice of the parameters γ and β is made based on the results of the simulations trying to achieve as few as possible number of dismissed blocks and at the same time trying not to waste so much RPs. The use of an excessive number of RPs will the effect of load the network unnecessarily. Here also two cases are considered when the controller is tuned.

- For the first situation, where the number of LPs per block is constant, the values γ and β are increased starting from 0 until the header value exceeds the constant number of LPs per block and remains stable. The obtained values are $\gamma = 170$ and $\beta = 89$.
- For the case of variable number of LPs in a block the scenario chosen to tune the ADSICM module is the same that was chosen for the ADICM module, a lossy link with uniform probability distribution and loss probability equal to 0.03.

5. RESULTS AND DISCUSSIONS

The AFEC system is implemented and simulated using Network Simulator 2 (NS-2) [23] and the NAM output is shown in Figure 10 (a) and 10 (b) below. The system parameters required for simulation are set as shown in Table 1. Analysis is carried out by number of blocks sent and number of packets sent. The performance of the three ICMs of AFEC sender as described in section 3, is evaluated using NS-2 and various results are plotted as shown in Figure 11(a)-11(h). All the simulations for the three ICMs have been running separately and afterwards all the results have been summarized and plotted together.

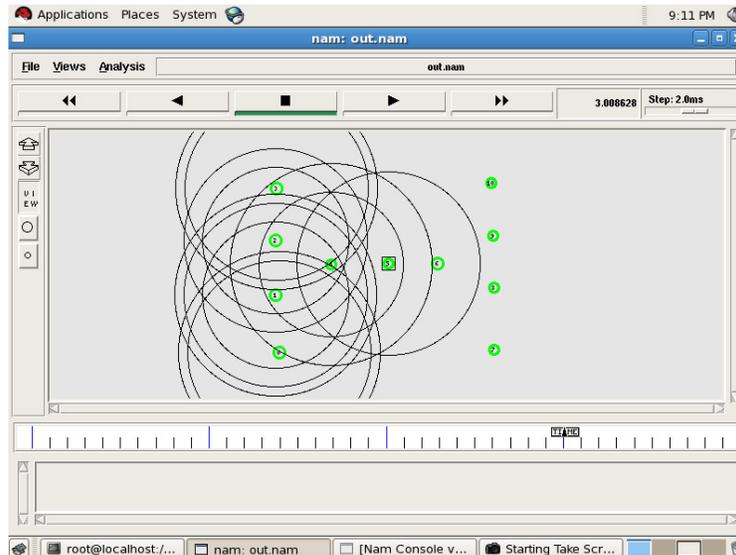


Fig. 10 (a) NAM Output-Wireless System Scenario

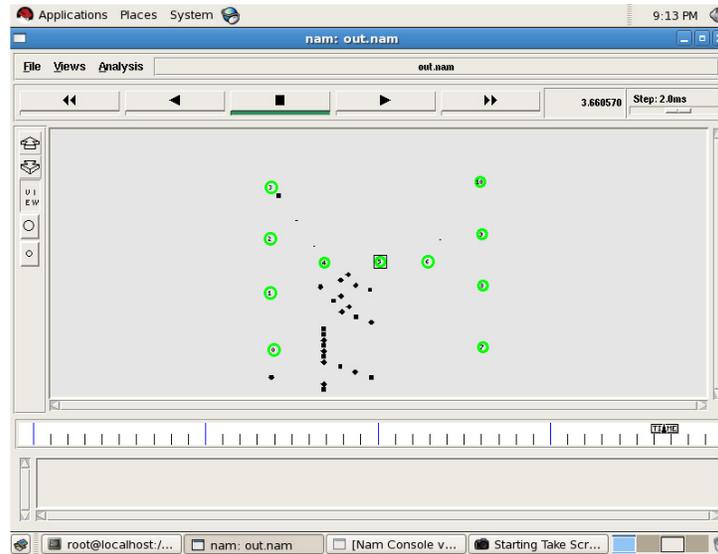


Fig. 10 (b) NAM Output-Video Transmission Scenario

Table 1. Simulation parameters

Parameter	Value
Packet Size	200 bytes
Packet Interval	0.004 second
Loss Rate for Error model	0.1 packets/sec
Number of Data blocks k	16
Gain γ for ADICM	170
Gain γ for ADSICM	170
Gain β for ADSICM	89

Now the scenario of congestion in nodes of wireless link is considered. The DBP is set to 0.005 for all the three suggested module of ICMs in AFEC. Figure 11 (a) shows the EBP which

is calculated from CSI and added to the DBP to generate error 'd'. This error 'd' is used by the FEC encoder to calculate the number of RPs based on the number of blocks 'k' of video data. The AFEC algorithm has to rectify this error after receiving the CSI from receiver by adapting the RPs accordingly. The simulation result shows that for DICM this error remains for all the blocks of data, for ADICM the error present only upto 25 blocks of video data whereas ADSICM rectifies the error within 10 blocks of data when transmission starts. Hence it is efficient.

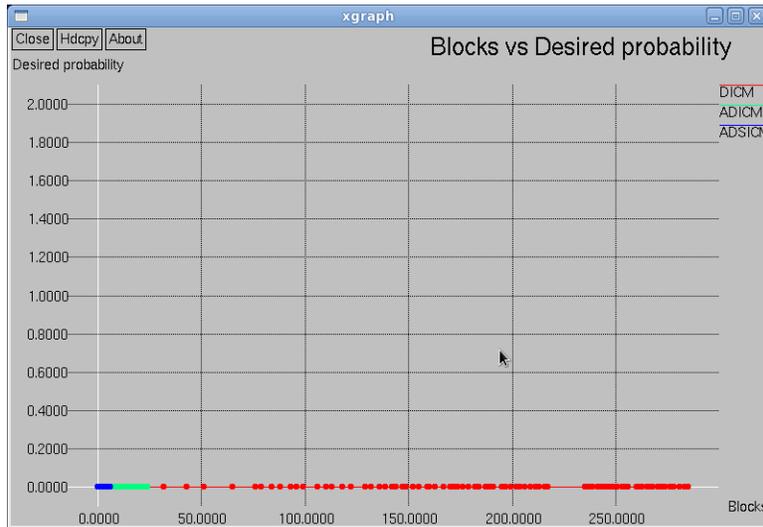


Fig. 11 (a) Number of Blocks Vs DBP

Figure 11 (b) shows the EBP which is calculated from CSI and added to the DBP as shown in Figure 5 to generate error 'd'.

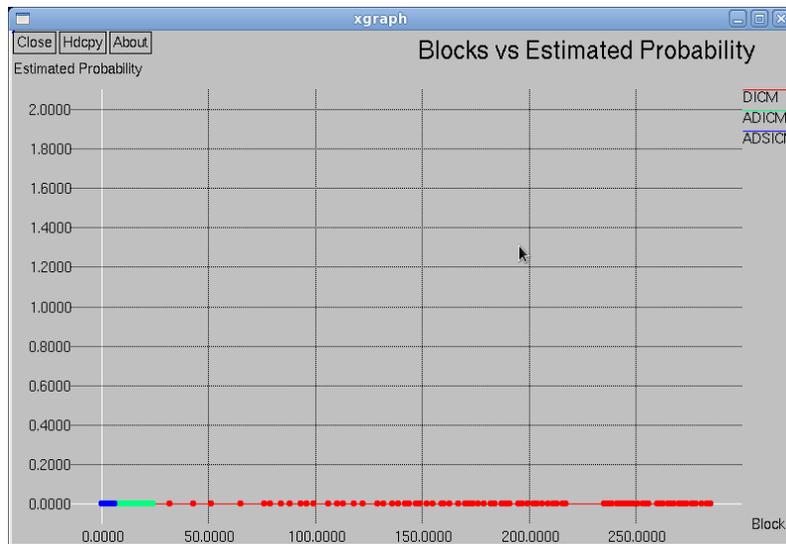


Fig. 11 (b) Number of Blocks Vs EBP

Figure 11 (c) indicates the relation between the blocks and the error introduced due to congestion in the wireless nodes. If error is present it is indicated by binary value '1' otherwise it will be '0'. The AFEC algorithm has to rectify this error after receiving the CSI from receiver by adapting the RPs accordingly. The simulation result shows that for DICM this error remains for all the blocks of data, for ADICM the error present only upto 25 blocks of video data

whereas ADSICM rectifies the error within 10 blocks of data when transmission starts. Hence it is efficient.

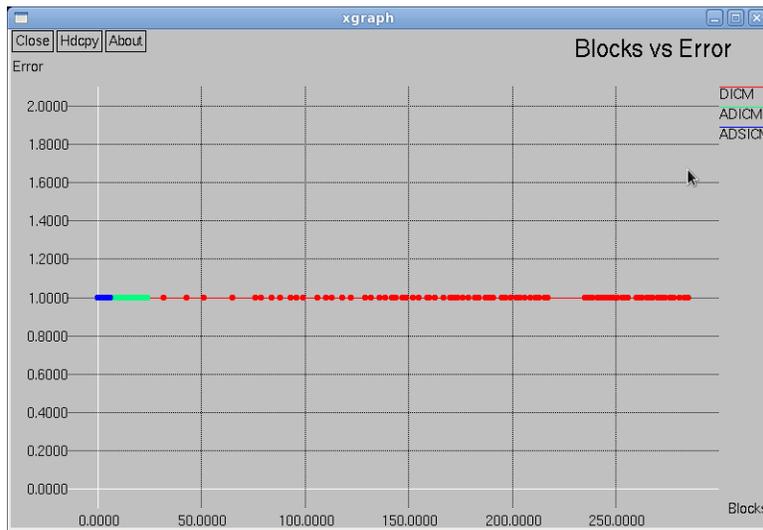


Fig. 11 (c) Number of Blocks Vs Error

Based on the EBP, the ICMs computes the RPs header size. These RPs are added to the data blocks to generate 'n' packets to be sent. Figure 11 (d) shows the size of HPs required by all the three ICMs. Large header size is required by the ADSICM but during the starting phase only which is used to find the error and rectify it. Introduction of integrator in ADSICM reduces EBP further to negligible value, because ADSICM reacts to the increase in the block loss probability by increasing the number of HPs as shown in result. The two other ICMs needs HP to be changed for more number of blocks of data still after the communication starts.

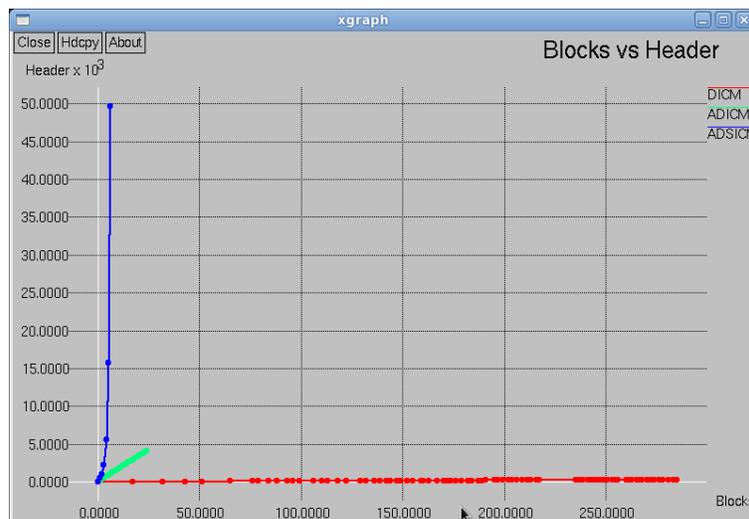


Fig. 11 (d) Number of Blocks Vs Header

Figure 11 (e) shows the relation between number of blocks of video data and loss probability. The packets are lost due to congestion in wireless channel. It is more clear that for ADSICM the packet loss probability is very less. Within 10 blocks of transmission itself it exactly computes the loss probability and receiver receives the data without error.

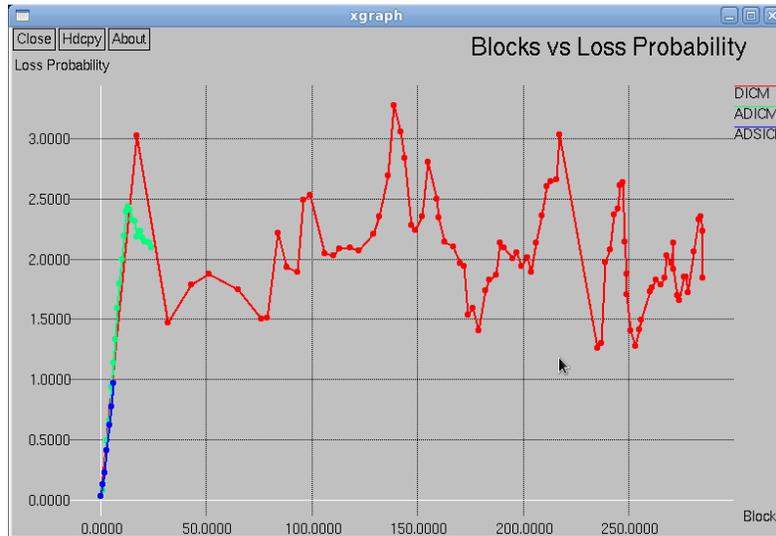


Fig. 11 (e) Number of Blocks Vs Loss Probability

The relation between the number of packet lost and the number of blocks of video data is shown in figure 11 (f). Number of packet lost within the block is avoided within 10 blocks in ADSICM, within 25 blocks in ADICM whereas errors are present in all data blocks in DICM causing more packet loss.

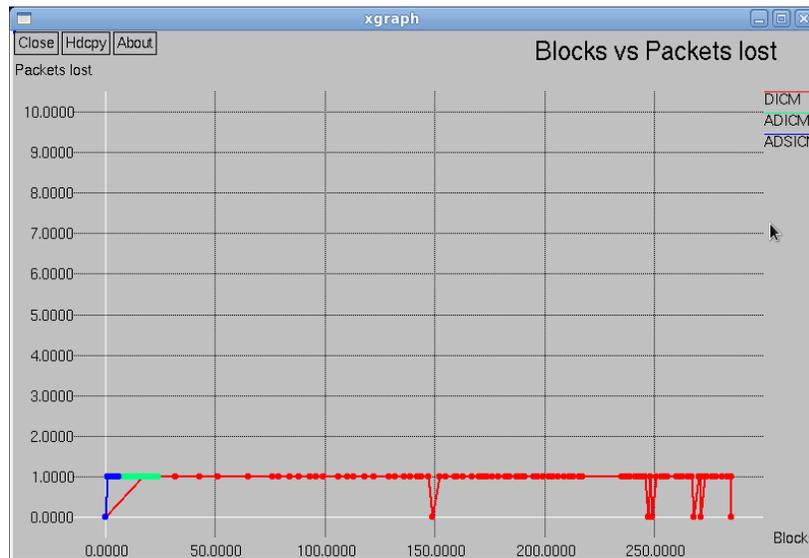


Fig. 11 (f) Number of sent Blocks Vs Packet Loss

Now the scenario of lossy wireless link is considered. The DBP for all the three ICMs is kept constant to 0.005 as shown in figure 11 (g). Figure 11 (h) shows the EBP. The error 'd' is used to adapt RPs value to rectify the errors of transmission. It is observed from figure 11 (i) that the errors are rectified by ADSICM during 25 x 10³ packets only after the transmission starts. The DICM and ADICM needs more number of packets to rectify the errors.

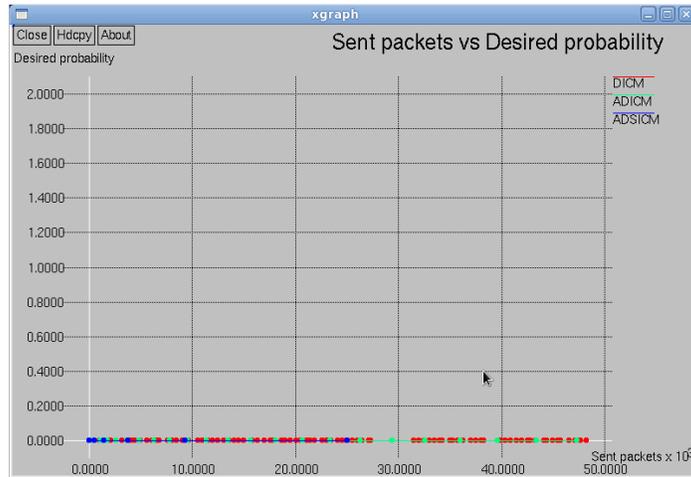


Fig. 11 (g) Number of sent packets Vs DBP

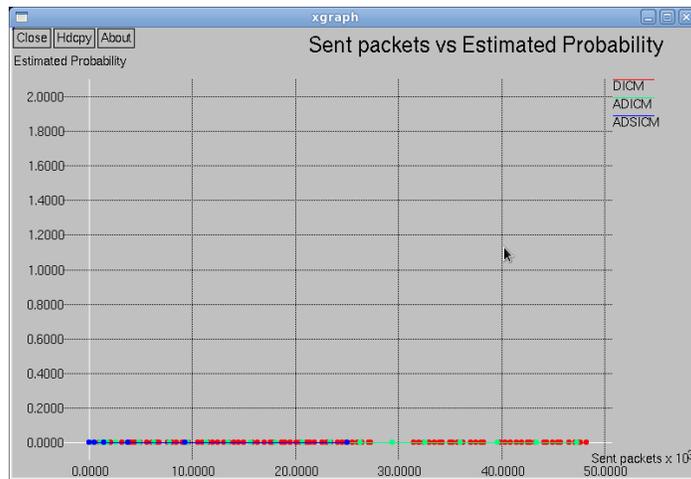


Fig. 11 (h) Number of sent packets Vs Estimated Probability

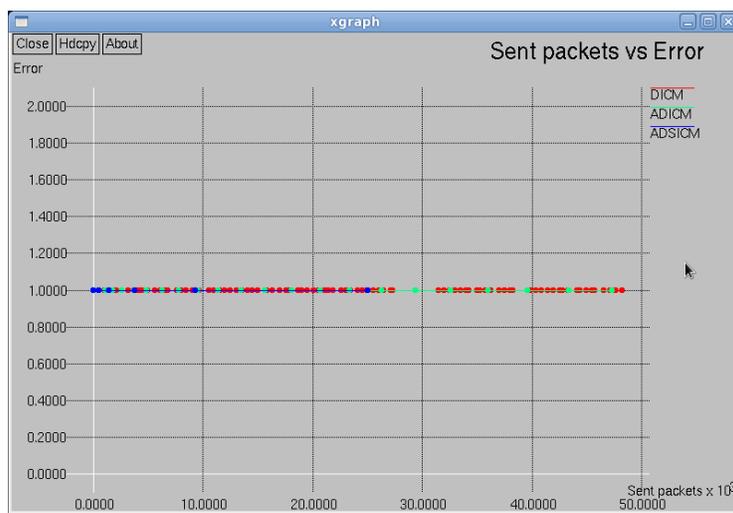


Fig. 11 (i) Number of sent packets Vs Error

Figure 11 (j) shows the size of HPs required by all the three ICMs. Large header size of 50×10^3 is required by the ADSICM but during the starting phase only i.e. upto 25×10^3 sent packets, which is used to find the error and rectify it. Introduction of integrator in ADSICM reduces EBP further to negligible value, because ADSICM reacts to the increase in the block loss probability by increasing the number of HPs in the sent packets as shown in result. Other two ICM needs HP to be added for more number of sent packets.

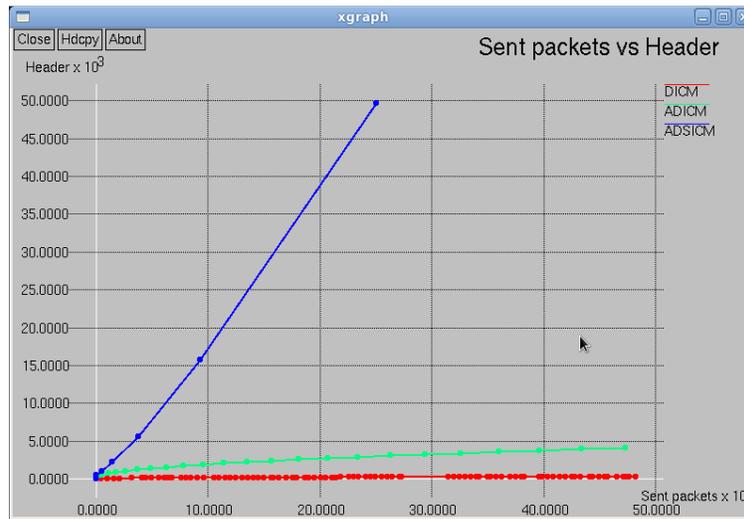


Fig. 11 (j) Number of sent packets Vs Header

The RPs header size has to be adaptive to the CSI received from the receiver. The CSI indicates packet loss probability. If packet loss probability is more, the header size has to be adaptive to this and intern it should increase. When the loss probability is zero the header size also has to be reduced. It adds additional parity information among the packets which will reduce the packet loss. The packet loss probability in a critical link is more for DICM, ADICM and reduces drastically for ADSICM as shown in Figure 11 (k) below.

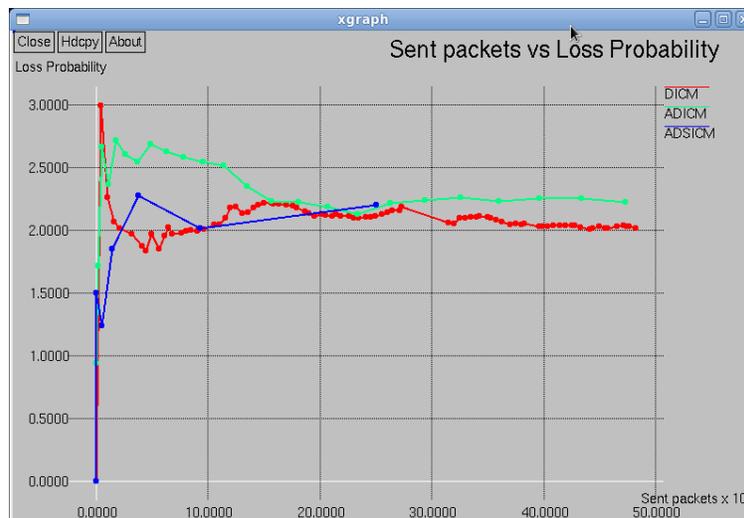


Fig. 11 (k) Number of sent packets Vs Loss Probability

The channel loss probability is kept constant to 0.03 and uniformly distributed. At the receiver side the LPs in a sent packets can be recovered by decoding if no more than the amount of introduced RPs has been lost in that block.

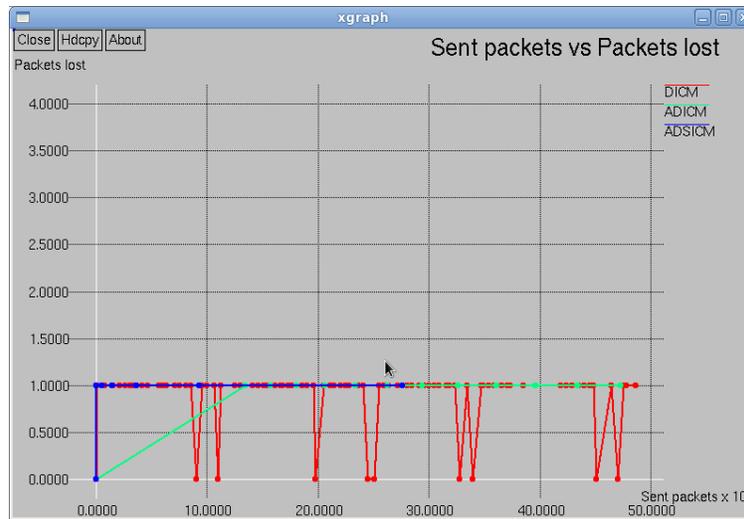


Fig. 11 (l) Number of sent packets Vs Packet Loss

Figure 11 (l) indicates the number of LPs for constant channel loss probability. As the constant loss probability is introduced in the channel, the sent packets are lost. The number of LPs for DICM is stopped at 48×10^3 sent packets, for ADICM it is stopped at 46×10^3 sent packets, but stopped at significantly lower value of sent packets for ADSICM, typically 28×10^3 as shown in figure 11 (l) below.

When examining throughput, the term 'Maximum Throughput' is frequently used. For all the three proposed ICMs, the maximum throughput is calculated and plotted as shown in figure 11 (m).

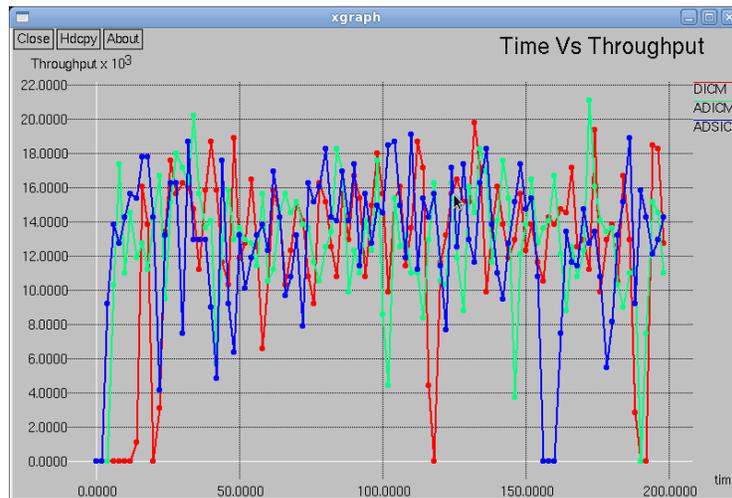


Fig. 11 (m) System Throughput

As seen in Figure 11 (m), maximum throughput for DICM is 12.1154×10^3 , that for ADICM is 12.6764×10^3 and for ADSICM it is coming to be 12.7358×10^3 . Thus, the algorithm of proposed ADSICM gives maximum throughput when transmitting a video over noisy or unreliable wireless channel.

6. CONCLUSIONS

Supporting video communication over lossy channels such as wireless networks is a challenging task due to the stringent QoS required by video applications and also affected by many channel impairments. The suggested ADSICM for real-time video transmissions can provide a much better end-user experience than existing approaches. The end to end delay is found to be 0.125169 seconds for ADSICM for the given loss rate of 0.3 % which is much lower. Also, ADSICM automatically modifies the quantity of RPs in FEC encoded packets through CSI by which reduces the number of packets lost significantly by increasing PDR to 99.7974 % for the loss rate of 0.3 % . Hence using ADSICM, transmission errors can be controlled to gain good QoS, so that the stability and good qualities of video transmission can be ensured. Using the suggested ADSICM, it is possible to improve network throughput of 12.7358×10^3 for reliable transmission of video over noisy and lossy channels. Thus, the suggested ADSICM for real-time video transmissions can provide a much better end-user experience than existing approaches.

REFERENCES

- [1] S. M. Koli, Radhika Purandare, S. P. Kshirsagar and V. V. Gohokar, "A Survey on Real time multimedia signal using Wireless Technology", Advances in Networks and Communications, Proc. of first International Conference on Computer Science and Information Technology, CCSIT 2011, Bangalore, India, January 2-4, 2011, Volume 132, Part 1, 137-147, DOI: 10.1007/978-3-642-17878-8_15.
- [2] S. M. Koli et. al., "Impact of Bit Error on Video Transmission over Wireless Networks and Error Resiliency", 2011 IEEE International Conference on Image Information Processing (ICIIP 2011), Simlala, India, November 3 - 5, 2011, DOI : 10.1109/ICIIP.2011.6108871.
- [3] Ramjee Prasad and Marina Ruggieri, Technology Trends in Wireless Communications, Artech House universal personal communications series, London, 2003.
- [4] Teerawat Issariyakul and Ekram Hossain, Introduction to Network Simulator NS2, Springer science + business media, New York, USA, 2009.
- [5] Behrouz A. Forouzan, Data Communications and Networking, 3rd edition, McGraw Hill publication, India, 2007.
- [6] Kazem Sohraby; Daniel Minoli; Taieb Znati, Wireless Sensor Networks: Technology, Protocols, and Applications, John Wiley & Sons, Inc. publication, Hoboken, New Jersey, 2007.
- [7] J. I. Hall, Notes on Coding Theory, USA, 2003.
- [8] Zhang, Q., et. al., "End-to-End QoS for video delivery over wireless Internet," Proc. of the IEEE, vol. 93, no. 1, pp. 124-234, Jan. 2005.
- [9] W. A Lee, G. Chan, Q. Zhang, W. Zhu, and Y. Q. Zhang, "Optimal Allocation of Packet-Level and Byte-level FEC in Video Multicasting over Wired and Wireless Networks", IEEE Globecom'01, Nov. 2001
- [10] Yubo Tan, et. al., "A video transmission algorithm over the internet based on FEC and Kalman", Proc. of IEEE International Symposium on IT in Medicine and Education, pp. 263-267, 2008.
- [11] Yakubu s. Baguda, "Adaptive FEC Error Control Scheme for Wireless video Transmission", Proc. of the 12th International conference on Advanced Communication Technology, ICACT 2010, South Korea, pp. 565-569, 2010.
- [12] Mihaela van der Schaar; et. al., "Adaptive Cross-Layer Protection Strategies for Robust Scalable Video Transmission Over 802.11 WLANs", IEEE Journal on Selected Areas in Communications, vol. 21, no. 10, pp. 1752-1763, Dec. 2003.
- [13] Zhang yongjun, et. al., "Error Control and Recovery Technology in Real-time Media Stream

- Transmission,” J. of National University of Defence Technology, vol. 3, no. 6, pp. 75-76, 2003.
- [14] Hadi S. Aghdasi et. al., “An energy efficient and high quality digital video transmission architecture in wireless video based sensor networks”, Sensors 2008, 8, 4529-4559; DOI: 10.3390/s8074529.
- [15] Reza Dianat et. al., “Reliable Video Transmission Using Codes Close to the Channel Capacity”, IEEE Transactions On Circuits And Systems For Video Technology, vol. 16, no. 12, pp. 1550-1556, Dec. 2006.
- [16] Aggelos K. Katsaggelos et. al., “Advances in Efficient Resource Allocation for Packet-Based Real-Time Video Transmission”, Proc. Of IEEE, vol. 93, no. 1, pp. 135-147, Jan. 2005.
- [17] Hassan Mansour et. al., “Dynamic Resource Allocation for MGS H.264/AVC Video Transmission Over Link-Adaptive Networks”, IEEE Transactions On Multimedia, vol. 11, no. 8, pp. 1478-1491, Dec. 2006.
- [18] H.Zeng and J.Boyce, “Packet Coding Schemes for MPEG video over Internet and Wireless Networks”, IEEE WCNC, vol 1.3, pp 191-195, Sept. 2000.
- [19] Gang Ding, Halima Ghafoor and Bharat Bhargava, “Error resilient video transmission over wireless networks”, IEEE Workshop on Software Technologies for Future Embedded Systems, pp. 31-34, 2003.
- [20] H. Schulzrinne, and S. Casner, “RTP: A Transport Protocol for Real-Time Applications”, RFC3550, the Internet Society, 2003.
- [21] Bernd Girod and Niko Farber, “Feedback-Based Error Control for Mobile Video Transmission”, Proc. Of IEEE, vol. 87, no. 10, pp. 1707-1723, Oct. 1999.
- [22] Harun, N.Z.; Ghazali, O.; , "Enhancement on adaptive FEC mechanism for video transmission over burst error wireless network," Information Technology in Asia (CITA 11), 2011 7th International Conference on , vol., no., pp.1-6, 12-13 July 2011, doi: 10.1109/CITA.2011.5999532
- [23] <http://www.isi.edu/nsnam/ns/>
- [24] Shan Yang; Wei Song; Zhangdui Zhong; , "Packet-Level Performance Analysis for Video Traffic over Two-Hop Mobile Hotspots," Wireless Communications Letters, IEEE , vol.1, no.2, pp.137-140, April 2012, doi: 10.1109/WCL.2012.022412.110252.

Authors

S. M. Koli received his B.E. degree from the Department of Electronics Engineering at the Shivaji University, Kolhapur, Maharashtra, India in 1997 and his M. Tech. degree from the Department of Electronics and Telecommunication Engineering at Dr. Babasaheb Ambedkar Technological University, Lonere, Maharashtra, India in 2006. Currently he is a Ph.D. student in the Department of Engineering and Technology, at Sant Gadage Baba Amravati University, Amravati, Maharashtra, India. His research interests are mainly in the wireless domain and cross-layer design and optimization for wireless networks.



R. G. Purandare received her B.E. degree from the Department of Electronics and Telecommunication Engineering at the University of Pune, Pune, Maharashtra, India in 1987 and her M. Tech. degree from the Department of Electronics and Telecommunication Engineering at the Sikkim Manipal University, Sikkim, India in 2006. Currently she is a Ph.D. student in the Department of Engineering and Technology, at Sant Gadage Baba Amravati University, Amravati, Maharashtra, India. Her research interests are mainly in the wireless and mobile communication.



Dr. S. P. Kshirsagar received his B.E. degree from the Department of Electronics and Telecommunication Engineering at the University of Pune, Pune, Maharashtra, India in 1986 and his M. E. degree from the Department of Electronics and Telecommunication Engineering at the University of Pune, Pune, Maharashtra, India in 1990. He received Ph.D. degree from Liverpool John Moores University, UK, in 1994, where he received ODASSS scholarship award. He received UK sponsorship by Texas Instruments; UK and Brite-Euram Consortium, UNDP Fellowship award at University of Texas El-Paso, USA and national merit scholarship; at GCOEP, Pune, INDIA. His research interests are in automatic testing, microprocessor technology, parallel processing, and image processing.



Dr. V. V. Gohokar received her B.E. degree from the Department of Electronics and Telecommunication Engineering at Sant Gadage Baba Amravati University, Amravati, Maharashtra, India in 1991 and his M. S. degree from the Department of Electronics and Telecommunication Engineering at Bits Pilani, Rajasthan, India in 1997. She received Ph.D. degree from Sant Gadage Baba Amravati University, Amravati, Maharashtra, India in 2009. Her research interest is in video processing.

