Three-Dimensional Absorbing Markov Chain Model for Video Streaming over IEEE 802.11 Wireless Networks

Azfar Moid, *Student Member*, IEEE, and Abraham O. Fapojuwo, *Senior Member*, IEEE

**Abstract — *The varying wireless channel conditions necessitate the use of error control mechanisms for reliable transmission of video streaming applications. Forward error correction (FEC) and automatic repeat request (ARQ) mechanisms are used at the data-link layer of IEEE 802.11 based wireless networks to avoid and recover from the channel errors. In this paper, a three-dimensional absorbing Markov chain model is presented to accurately calculate the packet transmission time when both the FEC and ARQ mechanisms are used. Based on the calculated packet transmission time and given maximum number of transmission attempts, the number of redundant FEC packets is adjusted to achieve an optimum tradeoff between network overhead and delay. Numerical results show that the three-dimensional absorbing Markov chain model accurately captures the packet delivery dynamics for a given maximum number of transmission attempts at the data-link layer. With the knowledge of accurate packet transmission time, the FEC parameter is adjusted to achieve higher quality video at the terminal devices. The adjustment of the number of FEC packets based on the proposed three-dimensional model brings the combined advantages of reduced network overhead and enhanced video quality*[1].

Index Terms — ARQ, FEC, IEEE 802.11, Markov chain model, Video streaming, Wireless networks.

I. INTRODUCTION

The advances in efficient video compression technologies have made it possible for the video applications to be transmitted over bandwidth constrained wireless channels. H.264 is the latest state-of-the-art international video coding standard developed by the joint video team (JVT) of ITU-T and ISO/IEC [1]. Because of its excellent compression efficiency and ability to transcode for the consumer electronic devices [2], H.264 standard showed a huge potential in video encoding market for wireless streaming.

Despite the advances in efficient transcoding mechanism, the impairments in wireless channel require error control techniques at the data-link layer. If carefully designed, forward error correction (FEC) and automatic repeat request (ARQ) mechanisms provide some protection against those channel errors. Hybrid FEC and ARQ schemes can be used in IEEE 802.11 based wireless video streaming networks to increase the overall system efficiency, enhance the video quality and maximize the usage of network resources. FEC is a double edge sword that must be applied with proper care because FEC trades bandwidth for latency to improve the loss rate. On the other hand, the ARQ mechanism is used to identify and possibly recover the missing packets, but with a penalty of increased delay due to retransmissions. When the FEC and ARQ mechanisms are employed for the video streaming applications, the time to transmit a data-link layer packet also increases. As video streaming applications are very sensitive to the delays incurred during a packet transmission, an accurate analytical model is required to capture the time dynamics of packet delivery when the FEC and ARQ algorithms are jointly applied at the data-link layer.

The current state-of-the-art approach for determining the average packet transmission time over lossy wireless channel is a two-dimensional absorbing Markov chain model presented in [3], where the authors have derived the expression for mean value of the packet transmission time. The approach presented in [3] is accurate for determining the packet transmission time when only FEC scheme is implemented. With the inclusion of data-link layer packet retransmission attempts, there is a requirement of recalculating the average packet transmission time. Some other recent works in the literature also argue for the implementation of ARQ schemes in conjunction with FEC. For example, how to reduce the failure probability of a basic ARQ protocol is studied in [4], in the context of data transmission over wireless channels, where it has been shown that the FEC, coupled with ARQ scheme, is efficient if the parameters of FEC and ARQ are carefully selected. Although the failure probability is an important metric, for video streaming applications, it is also important to calculate the average time to transmit a packet over a lossy wireless channel. Similarly, to dynamically adapt the variations of packet-loss level and quality of service (QoS) requirements, an adaptive hybrid FEC-ARQ scheme is presented in [5], where the authors developed a two-dimensional adaptive error-control scheme that dynamically adjusts not only the error-control redundancy, but also the code mapping structures. Although the flexibility to variations of different QoS requirements for the mobile multicast services is achieved

[1] A. Moid is with the Department of Electrical and Computer Engineering, University of Calgary, AB, T2N-1N4, Canada (email: amoid@ucalgary.ca).

A. O. Fapojuwo is with the Department of Electrical and Computer Engineering, University of Calgary, AB, T2N-1N4, Canada (email: fapojuwo@ucalgary.ca).

Contributed Paper
Manuscript received September 30, 2008

in [5] while keeping the transmission delays low, a quantitative analysis for the delay measurement is missing in their approach.

The motivation of this work comes from the fact that an analytical model to calculate the average packet transmission time over lossy wireless networks, when both the FEC and ARQ mechanisms are used, is missing in the literature. The analysis approach used in this paper is an extension of the two-dimensional Markov chain model presented in [3] for calculating the average packet transmission time over IEEE 802.11 based wireless networks. When the FEC scheme is supplemented by the ARQ mechanism, there is a requirement of updating the absorbing Markov chain model of [3] to capture the effects of ARQ over the average transmission time of a packet because every transmission attempt costs an additional round-trip time (RTT) delay, which is critical in video streaming applications and cannot simply be ignored. Intuitively, when the number of retries increases for a packet, the average transmission time also increases, hence the proposed model in this paper will quantify this increase in average packet transmission time. This accurate analysis is crucial in determining the new upper bound on the number of redundant FEC packets for a given maximum number of retransmission attempts.

The main contribution of this paper is twofold. First, it provides a three-dimensional absorbing Markov chain model to accurately calculate the average transmission time of a packet, over an IEEE 802.11 based wireless channel, when both the FEC and ARQ mechanisms are used. Second, based on the calculated transmission time, the number of redundant FEC packets at the data-link layer is also adjusted to reduce the probability of a packet not being used at the decoder because of violating the deadline time constraint, resulting in high video quality. Real-time channel conditions, probability of packet error, and packet transmission probabilities are used in the average transmission time calculation. The proposed model calculates the real-time parameters at the data-link layer and also provides the error-resilience functionality, based on the available processing time, which is crucial in designing the consumer electronics products.

The rest of the paper is organized as follows. In section II, we introduce the preliminaries of the analysis. Section III presents the three-dimensional absorbing Markov chain model developed to accurately calculate the average transmission time of a packet. In Section IV, numerical results are presented and compared with those of the existing two-dimensional absorbing Markov chain model, so that the effectiveness of the proposed model can be verified. Finally, this paper concludes in section V.

II. Preliminaries

A. Efficient Use of FEC and ARQ Schemes

Since a video sequence is composed of multiple video frames, throughout this paper, a packet refers to an IEEE 802.11 data-link layer data block unit, whereas a frame denotes a video frame at the application layer.

Before transmitting over an IEEE 802.11 based wireless network, a video frame of size G (bytes) is segmented into the N_{pd} packets, such that $\left\lceil N_{pd} = G/n \right\rceil$, where n is the packet size in bytes after a byte-level redundancy is added to it, and $\lceil x \rceil$ is the smallest integer greater than or equal to x. Assume that byte-level FEC combination m_b, and packet-level FEC combinations M_b, are used for the FEC implementation, such that, $m_b = (n, k_b)$ and $M_b = (N_p, N_{pd})$, respectively. Here, k_b is the original number of bytes per packet, such that $n = k_b + b$, and $b\,(0 \le b \le n-1)$ is the number of redundancy bytes added to a packet. Similarly, N_p is the total number of packets at the data-link layer of the given video frame, after redundancy is added. It can also be written as: $N_p = N_{pd} + N_{FEC}$, where N_{FEC} $(0 \le N_{FEC} \le N_{pd})$ is the number of redundant packets added to the given video frame. In our recent work [6], it is shown that by jointly designing the parameters of FEC and ARQ mechanisms and minimizing the cost-throughput-ratio (CTR) using the proposed dynamic programming technique, much of the lost information can be recovered without putting excessive load on the network. The accuracy of the presented model in [6] is dependent on two parameters: the variable deadline-time ($T_{deadline}$) at the byte-level and fixed RTT delay (T_{RTT}) at the packet-level. While the CTR based method meets the time constraint and calculates the number of redundant FEC units to be added at both the byte and packet levels, there is still a requirement of putting a cap on the number of packets to be transmitted to avoid the buffer overflow at the data-link layer. If at any given time, video traffic is not the only traffic present in the network, the limited size buffer of access point (AP) contains multiple packets of the different traffic flows. Buffer overflow occurs when the accumulated packets at the AP buffer cannot be transmitted in time, due to network congestion or bad channel condition. Compared to the other traffic flows, the nature of packets containing video data is different. If the total wait time of video packets in the AP buffer is more than their deadline time, such packets are useless. Intuitively, to avoid further congestion, either the number of packets containing video data be reduced, or the number of ARQ retransmission attempts be minimized. By default, most of the IEEE 802.11 based wireless device manufacturers set the value of number of transmission attempts, for a data-link layer packet, to be 1. The ARQ scheme suggests a limited number of retransmission attempts at the data-link layer to achieve a certain level of reliability for packet transmission. For the scenario where the maximum number of transmission attempts is restricted to 1 and no packet level FEC combination is used, the error recovery is left solely to the byte-level FEC mechanism, which has also shown considerable recovery from the channel errors [6].

B. Absorbing Markov Chain

A state s_a of a Markov chain is absorbing if it is impossible to leave that state once entered. A Markov chain is absorbing if it has at least one absorbing state and it is possible to go to an absorbing state from every other state, not necessarily in one step. Conversely, a state which is not absorbing is called a transient state. For an arbitrary absorbing Markov chain, if there are α absorbing states and β transient states, the transition matrix will have the following canonical form [7]:

$$\mathbf{\Psi}=\begin{bmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{R} & \mathbf{Q} \end{bmatrix}, \tag{1}$$

where $\mathbf{I}$ is a α-by-α identity matrix, $\mathbf{0}$ is a α-by-β zero matrix, $\mathbf{R}$ is a nonzero β-by-α matrix, and $\mathbf{Q}$ is a β-by-β matrix.

Following properties of the absorbing Markov chain model are used in this work:

1) The Fundamental Matrix

For an absorbing Markov chain $\mathbf{\Psi}$ the matrix $\mathbf{N} = (\mathbf{I} - \mathbf{Q})^{-1} = [n_{ab}]$, called the fundamental matrix of $\mathbf{\Psi}$, exists. The entry n_{ab} of $\mathbf{N}$ gives the expected number of times that the process is in the transient state s_b given it starts in the transient state s_a.

2) Time to Absorption

If the chain starts in state s_a, the expected number of steps before the chain is absorbed is a finite value. Given that the chain starts in state s_a, and let $\mathbf{t}$ be the column vector whose i^{th} entry is t_i, the vector $\mathbf{t} = \mathbf{Nc}$ gives the required value of t_i, where $\mathbf{c}$ is a column vector whose entries are all 1's.

III. Analysis of Average Packet Transmission Time

The packet transmission time, defined as the time between the first transmission attempt of a packet and its successful transmission, is a random variable and depends on multiple factors, such as, the maximum number of retransmission attempts allowed, current channel and network conditions, available channel bandwidth, and so on. For video streams, the transmission time of a packet must be less than its deadline time, otherwise the packet is considered useless and usually discarded by the decoder. In this section, we develop a model to analyze the average packet transmission time. Based on the calculated average transmission time, an efficient mechanism is required at the data-link layer to calculate the optimum number of FEC redundant packets, such that none or all of the video packets required to regenerate a video frame can be transmitted in a timely manner. The deadline time of a packet is directly proportional to the existing number of packets in the receiver queue and varies according to the structure of the decoder.

In the following, a three-dimensional absorbing Markov chain model is developed to calculate the average transmission time of a packet over a lossy wireless channel. The rationale for choosing the absorbing Markov chain model is based on two properties of such a model that apply to the problem under study. First, the fundamental matrix $\mathbf{N}$ for the problem under study exists and can be determined from knowledge of packet transition probabilities which in turn are obtained by writing the $\mathbf{\Psi}$ matrix. The second property that time to absorption be finite implies the existence of an absorption state, demonstrated as follows. For video packet transmission, if a packet is successfully transmitted, there is a zero probability of sending the same packet again over the network – this corresponds to the absorbing state. Further, a packet reaching its maximum number of retransmission attempt is also discarded, giving a zero probability of transmitting the same packet again. In general, the first state of each packet's transmission attempt would be the absorbing state of all the states of the previous packet.

A. Model Assumptions

Following are the assumptions made for model development:

A1. For consistency with Reference [3], a fast channel is assumed in which the channel conditions may change during the transmission of a data-link layer packet.

A2. Without loss of generality, an IEEE 802.11b based wireless local area network (WLAN) is considered, operating in a distributed coordination function (DCF) mode.

A3. For the ARQ scheme, the maximum number of transmission attempts per packet at the data-link layer is limited to R_{max}, so as not to exceed the tolerable delay limit ($T_{deadline}$) for a packet.

A4. Two levels of FEC implementation are considered: first, at the byte-level, and second, at the packet-level, as given in [6], to achieve higher reliability.

A5. Reed-Solomon (RS) error-correction code is used to generate the redundant FEC packets. The RS codes are well suited for wireless applications where errors occur in bursts.

A6. Stop-and-wait ARQ mechanism is used at the data-link layer, which sends a new packet only after getting a positive ACK from the receiver.

A7. Two channel states are assumed, good and bad. To account for channel uncertainties, it is assumed that there is a small probability of transmission error even when the channel is in a good state, and similarly there is a small probability of correct packet transmission in the case of bad channel condition.

A8. The decoder buffer holds a cushion of packets, which makes the deadline time of an incoming packet independent of the existing number of packets in the decoder queue.

B. Model Development

The proposed absorbing Markov chain model is three-dimensional because it relies on three major parameters, which are: 1) number of packets (in a video frame) remaining to be transmitted, 2) channel condition (good or bad), and 3) number of retries at the data-link layer. The state of the Markov chain is described as (i,j,k), where i denotes the number of packets remaining in a frame to be transmitted $(i \le N_p)$, j is the channel state $j \in \{0,1\}$ – where 0 stands for bad state and 1 represents good state, and k is the number of transmission attempts $(k \le R_{max})$.

The channel is modeled by a two state transition diagram, whose transition probabilities are shown in Fig. 1, and defined as follows:

P_{00} – probability of going from bad state to bad state,
P_{01} – probability of going from bad state to good state,
P_{10} – probability of going from good state to bad state,
P_{11} – probability of going from good state to good state.
Here, P_{11}=1-P_{10} and P_{00}=1-P_{01}.

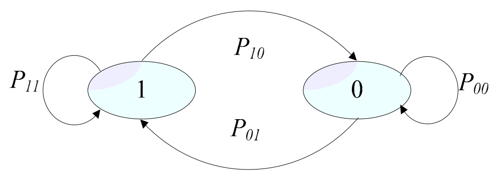


Fig. 1. Channel transition probabilities, where 0 represents the bad state, 1 represents the good state

Following the naming convention as used in [3], the packet transmission probabilities are described as: P_{c0} and P_{c1} – probabilities that a packet can recover from transmission errors when the channel is in bad and good state, respectively, where $P_{c0} < P_{c1}$. Furthermore, P_{f0} and P_{f1} are the probabilities of a packet not recovering from the transmission errors when the channel is in bad and good state, respectively, where $P_{f1} < P_{f0}$. It can also be written as: $P_{cj} = 1 - P_{fj}$, where $j \in \{0,1\}$ and represents the channel state. From assumption A4 and, based on CTR based model from our recent work [6], P_{fj} is given as:

$$P_{fj} = \sum_{q=N_p-N_{pd}+1}^{N_p} \binom{N_p}{q} \left[\Pr(n,k_b).(1-P_d)\right]^{N_p-q} . \left[1-\Pr(n,k_b).(1-P_d)\right]^{q} . \qquad (2)$$

Here, q is the number of erroneous packets which are not recoverable by packet-level FEC and contribute to the permanent loss of a video frame, and P_d is the probability of a packet loss due to limited data-link buffer size. Moreover, Pr(n,k_b) is the probability that a (n,k_b) FEC code is successful at the byte-level (i.e., at most n-k_b bytes are in error). Let ε denote the bit error rate (BER) of the channel and, assuming bit errors occur randomly and independently, the byte error rate (ρ) can be calculated as: ρ=1-(1-ε)^8. Finally, Pr(n,k_b) is calculated as [6]:

$$\Pr(n,k_b) @ \sum_{\upsilon=k_b}^{n} \binom{n}{\upsilon} (1-\rho)^{\upsilon} \rho^{n-\upsilon} . \qquad (3)$$

The packet transmission probability in transmission attempt k is denoted by P_{tk}, where k is the number of transmission attempts $(1 \le k \le R_{max})$. Note that with every unsuccessful packet transmission attempt, the number of packets residing at the AP buffer increases because of the continuous flow of incoming packets, resulting in increased probability of packet dropping at the AP queue. Thus, the probability of packet transmission during the (k+1)$^{\text{th}}$ attempt can be written recursively as: $P_{t(k+1)} = B_f P_{tk}$, where B_f $(0 \le B_f \le 1)$ is the data-link layer buffer occupancy.

Using the analogy of [3], the proposed three-dimensional absorbing Markov chain can be broken down into N_p identical stages. From Fig. 2, it is seen that the first data-link layer packet of a video frame, the Markov chain state (i,j,k) becomes (N_p,j,k). Similarly, the state (N_p-1,j,1) would be the absorbing state of all the (N_p,j,k) states, and so on. The final states (0,j,1) are the absolute absorbing states of a given video frame. A single stage is solved for calculating the average time to transmit a video packet ($T_{av,i,j}$) over the wireless network, where $T_{av,i,j}$ represents the average transmission time of a packet i for channel condition j. The total average time to transmit a video frame ($T_{tot,j}$) can then be calculated as:

$$T_{tot,j} = \sum_{i=1}^{N_p} T_{av,i,j} . \qquad (4)$$

Depending on the initial channel conditions, the Markov chain starts at state (N_p,j,1) for a given video frame. As the channel conditions may change during a data-link layer packet transmission (assumption A1), the next state would either be (N_p-1,j,1) or (N_p,j,2), for a successful or failed transmission attempt, respectively. For the packet N_p, the probabilities of going from one state to another are shown in Fig. 2. If the packet is successfully transmitted, the transition probabilities can be written as: $P_{xy}P_{cj}P_{tk}$, where $x, y \in \{0,1\}$ and represent the channel conditions. Similarly, for a failed transmission attempt, the transition probabilities are given as: $P_{xy}P_{fj}$(1-P_{tk}). The packet N_p is discarded if it cannot be transmitted in R_{max} number of attempts. As a packet is either successfully transmitted or prepared for the next transmission attempt, there is no transition between the states: (i,0,k) and (i,1,k), as can be seen in Fig. 2.

The average time to successfully transmit a single packet over the wireless channel can be calculated by writing the probability transition matrix of one stage of the absorbing Markov chain and solving for the absorbing time. The canonical form of the probability transition matrix is shown in Fig. 3. Note that the transition matrix is of the form given in Eq. (1). For convenience, the originating and destination states are also respectively written on the left and upper sides of the matrix in Fig. 3.

As there are two absorbing states (N_p-1,j,1) for the packet N_p in states (N_p,j,k), the value of α becomes 2. Similarly, β would be equal to 2.R_{max}, which represents all the transient states of the packet N_p. The dimension of all the sub-matrices, of the matrix in Fig. 3, becomes: **I** (2x2), **0** (2x2.R_{max}), **R** (2.R_{max} x2) and **Q** (2.R_{max}x2.R_{max}), which is also consistent with the general canonical form of such matrices, given in section II.B.

Fortunately, the matrix manipulation and inversion becomes non-tedious when the number of maximum transmission attempts is a finite number. For example, in the case of IEEE 802.11 based wireless networks, R_{max} can be chosen as 3 (as described in [8] and [9]). After some matrix manipulation, the

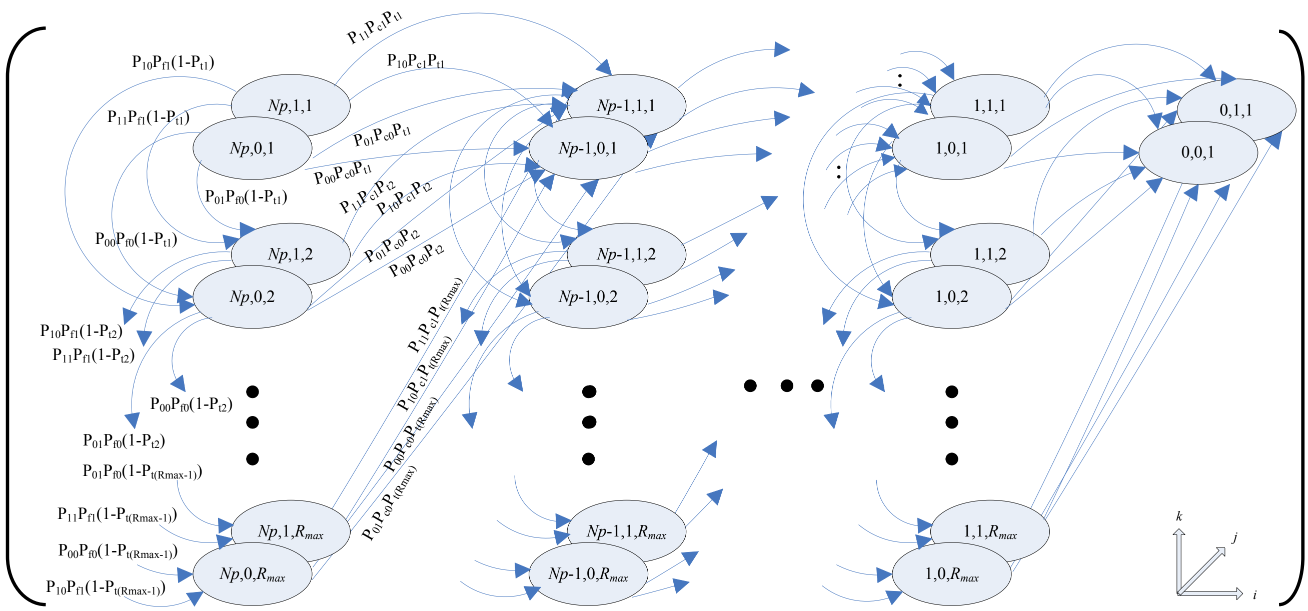


Fig. 2. Three-dimensional absorbing Markov chain model (*i* = number of packets, *j* = channel condition, *k* = transmission attempts)

	N_p-1,0,1	N_p-1,1,1	N_p,0,1	N_p,1,1	N_p,0,2	N_p,1,2	.	N_p,0,m	N_p,1,m	N_p,0,m+1	N_p,1,m+1	.	N_p,0,R_{max}-1	N_p,1,R_{max}-1	N_p,0,R_{max}	N_p,1,R_{max}
N_p -1,0,1	1	0	0	0	0	0	.	0	0	0	0	.	0	0	0	0
N_p -1,1,1	0	1	0	0	0	0	.	0	0	0	0	.	0	0	0	0
N_p,0,1	$P_{00}P_{c0}P_{t1}$	$P_{01}P_{c0}P_{t1}$	0	0	$P_{00}P_{f0}(1-P_{t1})$	$P_{01}P_{f0}(1-P_{t1})$	.	0	0	0	0	.	0	0	0	0
N_p,1,1	$P_{10}P_{c1}P_{t1}$	$P_{11}P_{c1}P_{t1}$	0	0	$P_{10}P_{f1}(1-P_{t1})$	$P_{11}P_{f1}(1-P_{t1})$	.	0	0	0	0	.	0	0	0	0
N_p,0,2	$P_{00}P_{c0}P_{t2}$	$P_{01}P_{c0}P_{t2}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0
N_p,1,2	$P_{10}P_{c1}P_{t2}$	$P_{11}P_{c1}P_{t2}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N_p,0,m	$P_{00}P_{c0}P_{tm}$	$P_{01}P_{c0}P_{tm}$	0	0	0	0	.	0	0	$P_{00}P_{f0}(1-P_{tm})$	$P_{01}P_{f0}(1-P_{tm})$	.	0	0	0	0
N_p,1,m	$P_{10}P_{c1}P_{tm}$	$P_{11}P_{c1}P_{tm}$	0	0	0	0	.	0	0	$P_{10}P_{f1}(1-P_{tm})$	$P_{11}P_{f1}(1-P_{tm})$	.	0	0	0	0
N_p,0,m+1	$P_{00}P_{c0}P_{tm+1}$	$P_{01}P_{c0}P_{tm+1}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0
N_p,1,m+1	$P_{10}P_{c1}P_{tm+1}$	$P_{11}P_{c1}P_{tm+1}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0
.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.	.
N_p,0, R_{max}-1	$P_{00}P_{c0}P_{t(Rmax-1)}$	$P_{01}P_{c0}P_{tRm-1}$	0	0	0	0	.	0	0	0	0	.	0	0	$P_{00}P_{f0}(1-P_{t(Rmax-1)})$	$P_{01}P_{f0}(1-P_{t(Rmax-1)})$
N_p,1, R_{max}-1	$P_{10}P_{c1}P_{t(Rmax-1)}$	$P_{11}P_{c1}P_{tRm-1}$	0	0	0	0	.	0	0	0	0	.	0	0	$P_{10}P_{f1}(1-P_{t(Rmax-1)})$	$P_{11}P_{f1}(1-P_{t(Rmax-1)})$
N_p,0, R_{max}	$P_{00}P_{c0}$	$P_{01}P_{c0}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0
N_p,1,R_{max}	$P_{10}P_{c1}$	$P_{11}P_{c1}$	0	0	0	0	.	0	0	0	0	.	0	0	0	0

Fig. 3. The general form of probability transition matrix including all absorbing and non-absorbing states for the first stage of delivering N_p packet

fundamental matrix (**N**) for R_{max}=3 is given in Fig. 4. Note that the dimension of **N** ($\mathbf{N} = (\mathbf{I}-\mathbf{Q})^{-1}$) is the same as that of **Q**, i.e., ($2.R_{max}$x$2.R_{max}$) or (6x6), when R_{max}=3. The probabilities denoted by *a, b, c, d, e, f, g, h* are calculated using the fundamental matrix **N**, and given in Table 1.

The mean time to transmit a single packet *i* over an IEEE 802.11 network, given the channel state $j \in \{0,1\}$, can be calculated as (using $\mathbf{t} = \mathbf{Nc}$):

$$T_{av,i,1} = \left[1(1+0)+2(-a-b)+3(a.e+b.g+a.f+b.h)\right]T_{RTT} \ , \ (5)$$

$$T_{av,i,0} = \left[1(0+1)+2(-c-d)+3(c.e+d.g+c.f+d.h)\right]T_{RTT} \ . \ (6)$$

To avoid the packet from being rejected at the receiver, $T_{tot,j}$ has to satisfy the time constraint of $T_{tot,j} \le T_{deadline}$, where $T_{tot,j}$ is given by (4), and $T_{deadline}$ is the deadline time of a given video frame. As the packets are of equal size for any given video frame, the deadline time of the frame is just the

$$\mathbf{N}=\begin{pmatrix} 1 & 0 & -a & -b & a.e+b.g & a.f+b.h \\ 0 & 1 & -c & -d & c.e+d.g & c.f+d.h \\ 0 & 0 & 1 & 0 & -e & -f \\ 0 & 0 & 0 & 1 & -g & -h \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix}$$

Fig. 4. The fundamental matrix for R_{max}=3

summation of deadline times of the packets in that frame.

C. Limiting FEC and ARQ count

From assumption A8, the deadline time is independent of the existing number of packets in the decoder queue, hence the time constraint $T_{tot,j} \le T_{deadline}$ can only be fulfilled by adjusting $T_{tot,j}$. Based on the calculated average transmission time of a video packet, it is proposed here to reduce the number of redundant FEC packets, to satisfy the above constraint. The pseudo code of the algorithm to enforce the time constraint is given in (7):

$$\begin{aligned} & while\ \left((T_{tot,j} > T_{deadline})\ \&\ (N_{FEC} > 0)\right) \\ & \{ \\ & \quad N_{FEC} = N_{FEC} - 1 \\ & \} \end{aligned} \tag{7}$$

This shows that the number of redundant FEC packets is reduced till the time constraint is met or the number of redundant FEC packets reduces to zero, whichever comes first.

An extension to the above algorithm is also developed by controlling the number of transmission attempts for each packet in conjunction with controlling the number of FEC packets. Formally, if the time constraint cannot be satisfied with the reduced number of FEC packets, the maximum retransmission limit should also be reduced to meet the time constraint. The pseudo code of the algorithm is given as:

$$\begin{aligned} & while\ \left((T_{tot,j} > T_{deadline})\ \&\ (N_{FEC} > 0)\right) \\ & \{ \\ & \quad N_{FEC} = N_{FEC} - 1 \\ & \} \\ & while\ \left((T_{tot,j} > T_{deadline})\ \&\ (R_{\max} > 1)\right) \\ & \{ \\ & \quad R_{\max} = R_{\max} - 1 \\ & \} \end{aligned} \tag{8}$$

The algorithm in (8) shows that if the time constraint is violated, the constraint can be satisfied by: 1) reducing the number of redundant FEC packets in a video frame, and 2) limiting the number of retransmission attempts.

IV. Results and Discussion

The proposed three-dimensional absorbing Markov chain model has been implemented in the JM reference software

TABLE I
Different values of the probability combinations

Value	Probability Combination
a	$P_{00}P_{f0}P_{t1}$ - $P_{00}P_{f0}$
b	$P_{01}P_{f0}P_{t1}$ - $P_{01}P_{f0}$
c	$P_{10}P_{f1}P_{t1}$ - $P_{10}P_{f1}$
d	$P_{11}P_{f1}P_{t1}$ - $P_{11}P_{f1}$
e	$P_{00}P_{f0}P_{t2}$ - $P_{00}P_{f0}$
f	$P_{01}P_{f0}P_{t2}$ - $P_{01}P_{f0}$
g	$P_{10}P_{f1}P_{t2}$ - $P_{10}P_{f1}$
h	$P_{11}P_{f1}P_{t2}$ - $P_{11}P_{f1}$

(ver. 13.2) [10]. The NS2 based platform [11], enhanced by Evalvid framework [12], is used for the simulation purpose. For the purpose of capturing realistic network operation conditions, three different sources are considered in the simulation environment. The first one is a video streaming source, the second source is a FTP source transmitting packets using the TCP protocol, and the third source is an exponential source transmitting packets using the UDP protocol. The FTP source represents a bulk file transfer application over the TCP protocol and, for the simulation, the file is considered big enough such that there is always data to transmit over the runtime of the simulation. The exponential source represents the bursty traffic, with a maximum packet size of 1500 bytes. Burst time and idle time are each set to 0.5 seconds. The source rate is set at 256 Kbps. Three different video sequences, i.e., *akiyo*, *container* and *foreman* are selected each from low, medium and high motion categories, respectively. First 100 frames of the three video sequences are encoded. The first frame of the video sequence is an intra (I-) frame, containing only the intra coded macro-blocks. Subsequent frames are predicted (P-) frames that allow both the intra coded and predicted macro-blocks. The video frame rate is set to 30 frames per second to emulate the television level picture quality. The rate-distortion (RD) optimization [1] was enabled and context adaptive binary arithmetic coding (CABAC) [1] was used for the entropy encoding. Joint FEC/ARQ, as described in our previous work [6], is implemented at the data-link layer, for error correction. Without loss of generality, an IEEE 802.11b link is selected between the access point (AP) and the client device, where the maximum data rate is 11 Mbps. It is important to note here that the IEEE 802.11a/g links merely provide higher data rate (i.e., 54 Mbps), without any different error correction mechanism that is not present in the IEEE 802.11b link. Consequently, if an IEEE 802.11 a/g link is chosen, the results will follow the same trends as those presented in this section.

Results for two-dimensional absorbing Markov chain model [3] are compared with those for the proposed three-dimensional absorbing Markov chain model, assuming a maximum retry limit (R_{max}) of 1, 2 and 3. As adaptive FEC scheme is used at the data-link layer to generate redundant FEC packets, the number of FEC packets generated for the good channel condition (when the probability of packet error ranges from 10^{-4} to 10^{-2}), is too few to be used for comparison. Therefore, only moderate to bad channel conditions are considered (when the probability of packet error ranges from 10^{-2} to 10^{0}). This gives a more realistic count of the redundant FEC packets required for comparison. The channel state transition probabilities are set to $P_{00} = 0.5$, $P_{01} = 0.5$, $P_{10} = 0.1$, and $P_{11} = 0.9$, representing the wireless channel has a tendency of being in the good state most of the time. The packet correction probabilities are considered as $P_{c0} = 0.8$ and $P_{c1} = 0.8$, showing that most of the packet errors are correctable using byte-level and packet-level FEC schemes. It is assumed that the buffer at the data-link layer of AP is large enough, such that there is no packet dropping, which further implies that the packet transmission probability becomes: $P_{t(k+1)} = P_{tk}$. Note also that the packet transmission probability varies inversely with the packet dropping probability. If the packet is transmitted in its first attempt, the effect of increasing R_{max} on the average frame transmission time will not be distinguishable, therefore an arbitrary low value of 0.1 is selected for P_{tk}.

A. Average Transmission Time of a Video Frame

The calculated average frame transmission time of the video sequences *akiyo*, *container*, and *foreman* are shown in Figs. 5a, 5b and 5c, respectively. When the first 50 video frames are compared, for the packet error probability of 0.3, it is seen that the average frame transmission time increases monotonically, when it is calculated by using the three-dimensional model, for the transmission limits (R_{max}) of 1, 2 and 3. This is due to the fact that, contrary to two-dimensional model, the three-dimensional model includes the effect of the number of transmission attempts at the data-link layer. Hence, a respective increase of 8 and 16 percent is observed, in frame transmission time, for R_{max} set to 2 and 3 in three-dimensional model, as compared to the two-dimensional model. When R_{max} is set to 1, representing only a single transmission attempt for each packet, the frame transmission time calculated from the three-dimensional model matches with that of the two-dimensional model, for all the three test video sequences. This not only validates the accuracy of the proposed three-dimensional model, but also shows the effect of R_{max} on the calculation of average packet transmission time. Moreover, as RD optimization is enabled for video encoding, the average frame size for the three video sequences is almost identical, which relates to the almost identical average transmission time for the three test video sequences (regardless of their motion categories), as can be seen in Fig. 5. An increase in frame transmission time, when three-dimensional absorbing Markov chain model is used, shows a poorer performance in terms of latency, as compared to the case where two-dimensional absorbing Markov chain model is used because the latter under-estimates the average transmission time. However, with this accurate knowledge of the frame transmission time, it is actually possible to adapt the underlying FEC algorithm to generate just enough redundant FEC packets, such that the time constraint $(T_{tot,j} \le T_{deadline})$ is met to avoid packet rejection at the decoder.

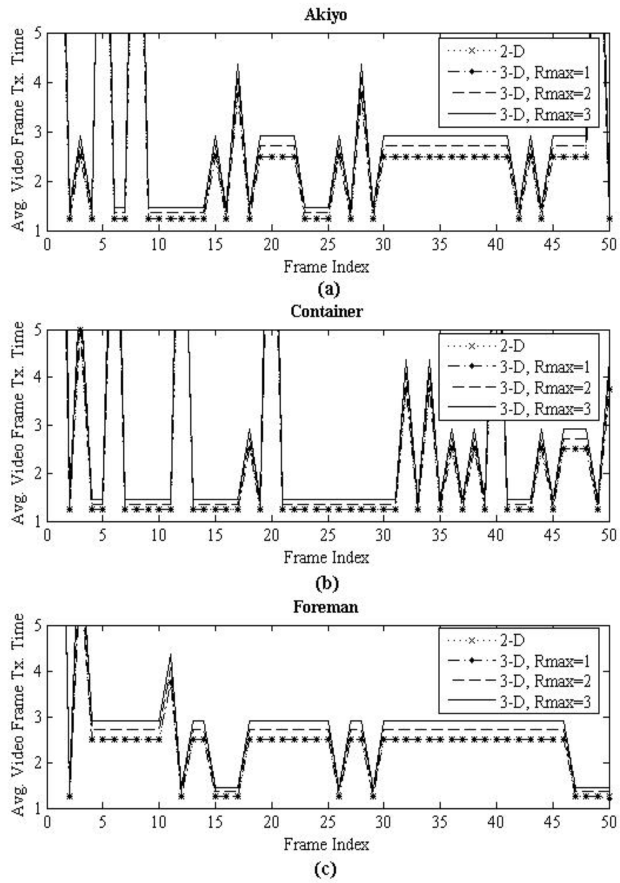
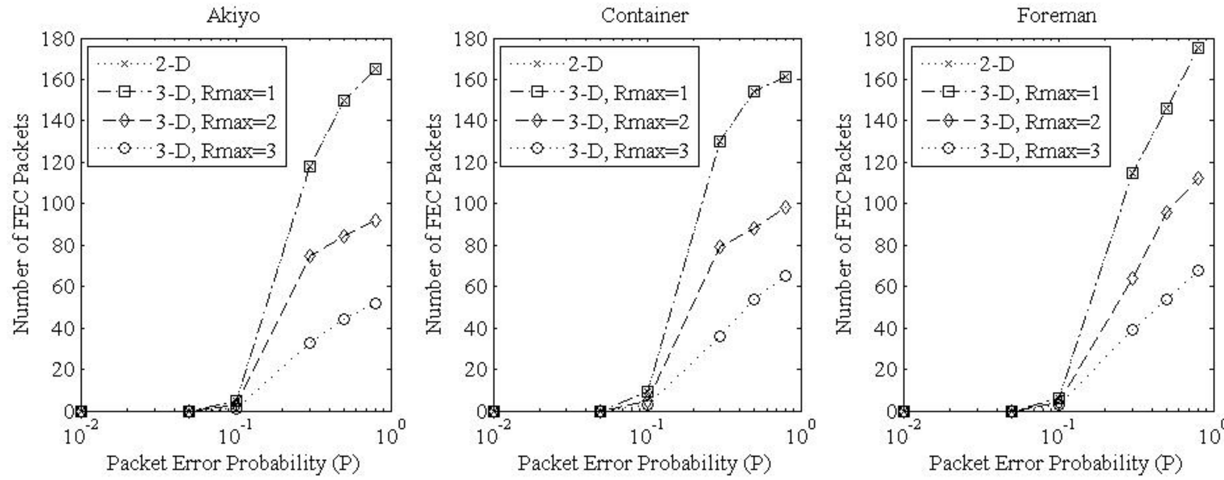


Fig. 5. Video frame transmission time for (a) *akiyo* (b) *container* and (c) *foreman*

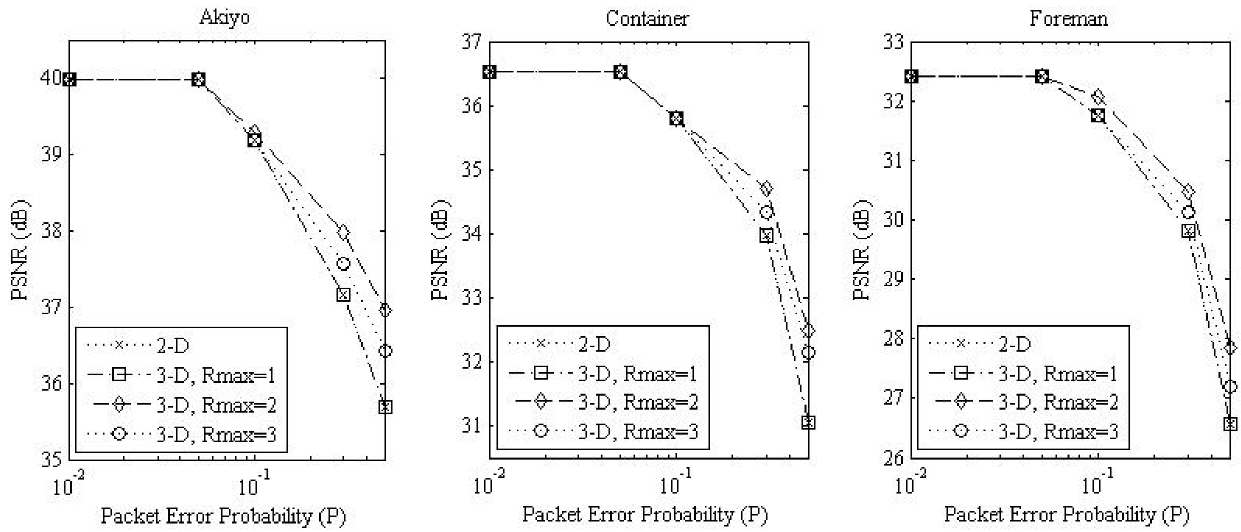
B. Number of Redundant FEC Packets

When the channel condition worsens (i.e., the probability of packet error goes from 10^{-2} to 10^{0}), the number of redundant FEC packets are accordingly increased to provide error resiliency against the bad channel condition. Because of the fixed deadline time (assumption A8), the time constraint $(T_{tot,j} \le T_{deadline})$ is satisfied by limiting the number of redundant FEC packets, for the given number of transmission attempts. This puts an upper bound on the number of redundant FEC packets to avoid any packet dropping at the decoder queue because of not meeting the deadline time constraint. The total number of FEC packets induced in the video stream, when the probability of packet error is varied from 10^{-2} to 10^{0}, is shown in Figs. 6a, 6b and 6c, for the three video sequences *akiyo*, *container* and *foreman*, respectively. As noted earlier, the three-dimensional model with R_{max} set to 1 is essentially the same as that of two-dimensional model, so an overlap of the two results is observed in Fig. 6. Further, two trends can also be seen in Fig. 6: 1) the number of FEC packet increases when the probability of packet error is higher and 2) the number of FEC packets generated when the two-dimensional model is used is more than the number of packets



(a) (b) (c)

Fig. 6. Number of redundant FEC packets generated for 100 frames of the video sequences: (a) *akiyo*, (b) *container* and (c) *foreman*



(a) (b) (c)

Fig. 7. PSNR improvement for the video sequences: (a) *akiyo*, (b) *container* and (c) *foreman*

generated when three-dimensional model (when R_{max} is set to 2 and 3) is used. For channels with high probability of error, there is a requirement of more FEC packets to compensate the channel errors, and hence an increased number of FEC packets are seen in Fig. 6 under bad channel condition. The reduced number of FEC packets in the case of three-dimensional model as compared to two-dimensional model avoids the violation of the time constraint at the decoder. For example, there is a reduction of 40 and 60 percent in the number of redundant FEC packets, when the value of R_{max} is set to 2 and 3, respectively, and the packet error probability is greater than 0.1. The reduction in the FEC packets translates to less overhead on the network. It is interesting to note that the reduction in the number of redundant FEC packets comes without any cost because the error recovery is still possible due to the byte-level FEC combination. It is concluded here that, with the accurate knowledge of the frame transmission time using three-dimensional absorbing Markov chain model, it is actually possible to adapt the FEC generation algorithm to generate just enough redundant FEC packets such that the total number of packets will conform to the time constraint.

C. *Achieved Video Quality*

Peak-signal-to-noise-ratio (PSNR) is used in this paper as the measure of objective video quality. As concluded from the previous subsection that the number of FEC packets is reduced to satisfy the time constraint, it can be seen in Fig. 7 that this reduction of redundant FEC packets is in fact useful in terms of overall video quality. Figs. 7a, 7b and 7c show the PSNR improvement for video sequences *akiyo*, *container* and *foreman*, respectively. When the channel condition gets worse (i.e., the probability of error goes beyond 10^{-1}), a marked improvement of over 1dB in PSNRs is observed in all the three test video sequences. This improvement is due to the fact that in the case of bad channel condition, the less number of FEC packets added per frame guarantees the non-violation of the deadline time constraint. The video frames rendered to the client device without being dropped causes the increase in

PSNR. As the RD optimization mode is enabled at the video encoder, the PSNR value for the video sequence with high motion content (e.g., *foreman* in this case) is the least as compared to the other video sequences. It is also seen in Fig. 7 that the PSNR value of two-dimensional model is exactly the same as that of the three-dimensional model when the value of R_{max} is set to 1, which is also consistent with our previous results. It is also observed in Fig. 7 that a maximum PSNR is achieved when R_{max} is set to 2. This is attributed to the fact that, for the P-frames where more macro-blocks are I-coded, the instantaneous size of the frame (in bytes) increases. Even without adding any FEC packets at the data-link layer, the probability of a frame being transmitted in the given deadline time is low, and a higher value of R_{max} just aggravates the problem. It is concluded here that with the accurate knowledge of the average frame transmission time, the number of redundant FEC packets can be reduced such that the time constraint is met, thereby increasing the video quality.

V. Conclusion

In this paper, we have presented a three-dimensional absorbing Markov chain model for estimating the average frame transmission time over an IEEE 802.11 based wireless channel. Based on the calculated frame transmission time, the number of redundant FEC packets is reduced to abide by the packet deadline time. Results show that up to 60% reduction can be achieved in the number of FEC packets, meaning less load on the wireless network. It is also found that the transport of FEC overhead does not degrade the video quality. Instead, simulation results show that the video quality increases by up to 1dB, when the three-dimensional absorbing Markov chain model is used to calculate the average frame transmission time, as compared to the case where two-dimensional model is used. This is because only a few video packets not meeting the delay constraints are dropped, when the three-dimensional absorbing Markov chain model is used, compared to the two dimensional model. Hence, the tradeoff presented by ARQ on the increase in frame transmission time actually reduces the probability of packets being rejected at the decoder side. It is concluded that the proposed model achieves enhanced video quality especially under bad channel conditions.

Acknowledgment

The authors acknowledge the support of the University of Calgary, TRLabs and National Sciences and Engineering Research Council (NSERC) Canada for this research.

References

[1] ITU-T and ISO/IEC JTC1, "Advanced video coding for generic audiovisual services," ITU-T Recommendation H.264 – ISO/IEC 14496 AVC, 2003.

[2] Z. Wei, K. L. Tang and K. N. Ngan, "Implementation of H.264 on mobile device," *IEEE Transactions on Consumer Electronics*, vol. 53, no. 3, Aug. 2007, pp. 1109 – 1116.

[3] M. Hassan and M. Krunz, "Video streaming over wireless packet networks: an occupancy-based rate adaptation perspective," *IEEE Transactions on Circuits and Systems for Video Technology*, vol.17, no.8, Aug. 2007, pp.1017-1027.

[4] A. Willig, "Redundancy concepts to increase transmission reliability in wireless industrial LANs," *IEEE Transactions on Industrial Informatics*, vol 1, no. 3, Aug. 2005, pp. 173 – 182.

[5] Xi. Zhang and Q. Du, "Adaptive low-complexity erasure-correcting code-based protocols for QoS-driven mobile multicast services over wireless networks," *IEEE Transactions on Vehicular Technology*, vol. 55, no. 5, Sep. 2006, pp. 1633 – 1647.

[6] A. Moid and A. O. Fapojuwo, "An analytical model for optimum byte-level and packet-level FEC assignment using buffer dynamics," *Research Letters in Communications*, Article ID 546184, 2008.

[7] C. M. Grinstead and J. L. Snell, "Introduction to probability," American Mathematical Society, Chapter 11, Second edition, ISBN-10: 0821807498, ISBN-13: 978-0821807491, July 1997.

[8] V. Sgardoni, P. Ferre, A. Doufexi, A. Nix and D. Bull, "Frame delay and loss analysis for video transmission over time-correlated 802.11a/g channels," *IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, PIMRC 2007. 3-7 Sept. 2007.

[9] V. Subramanian, K. K. Ramakrishnan and S. Kalyanaraman, "Disruption-tolerant link-level mechanisms for extreme wireless network environments," *2nd International Conference on Communication Systems Software and Middleware*, COMSWARE 2007. 7-12 Jan. 2007.

[10] JM Reference Software: http://iphome.hhi.de/suehring/tml/download/, accessed on July 14, 2008.

[11] Network Simulator, http://www.isi.edu/nsnam/ns/, accessed on July 14, 2008.

[12] Evalvid - A video quality evaluation tool-set, http://www.tkn.tu-berlin.de/research/evalvid/, accessed on July 14, 2008.

Azfar Moid (S'05) received his B.Sc. degree (with honors) in Electrical Engineering from the University of Engineering and Technology, Lahore, Pakistan, in 2000. He worked for Siemens Pakistan, in the department of Information and Communication Network (ICN) Services, where he was a member of technical support team. He got the scholarship from the Government of Finland to study his Masters degree in Telecommunication Engineering at Asian Institute of Technology (AIT), Thailand. He got his M.Engg. degree in 2001, and his thesis was awarded 'Excellent'. From 2003 to 2005, he worked as a Research Associate in the Space Technology Application and Research (STAR) program of AIT, where he was involved in the state-of-the-art research technologies in remote sensing and space program. Since 2005, he is a PhD student at the University of Calgary, Canada, in the department of Electrical and Computer Engineering. His current research interests are: protocol design for wireless video streaming and cross-layer signaling mechanisms.

Abraham O. Fapojuwo (S'85, M'89, SM'92) received the B.Eng. degree (first class honors) from the University of Nigeria, Nsukka, in 1980 and the M.Sc. and Ph.D. degrees in Electrical Engineering from the University of Calgary, Calgary, AB, Canada, in 1986 and 1989, respectively. From 1990 to 1992, he was a Research Engineer with NovAtel Communications Ltd., where he performed numerous exploratory studies on the architectural definition and performance modeling of digital cellular systems and personal communications systems. From 1992 to 2001, he was with Nortel Networks, where he conducted, led and directed system-level performance modeling and analysis of wireless communication networks and systems. In January 2002, he joined the Department of Electrical and Computer Engineering, University of Calgary, as an Associate Professor. He is also an Adjunct Scientist at TRLabs, Calgary. His current research interests include protocol design and analysis for future generation wireless communication networks and systems, and best practices in software reliability engineering and requirements engineering. Dr. Fapojuwo is a registered Professional Engineer in the Province of Alberta.