IMPROVING UDP PERFORMANCE USING INTERMEDIATE QoD-AWARE HOP SYSTEM FOR WIRED/WIRELESS MULTIMEDIA COMMUNICATION SYSTEMS

Khalid Darabkh[1] and Ramazan Aygün[2*]

[1]*Computer Engineering Dept, The University of Jordan, Amman 11942*

[2]*Computer Science Dept.*

University of Alabama in Huntsville

Huntsville, AL 35899

Phone: 1-256-8246455

Fax: 1-256-8246239

Email: k.darabkeh@ju.edu.jo, raygun@cs.uah.edu

* Corresponding Author

ABSTRACT

Multimedia communication in wireless networks is challenging due to the inherent complexity and constraints of multimedia data. To reduce the high bandwidth requirement of video streaming, videos are compressed by exploiting spatial and temporal redundancy thus yielding dependencies among frames as well as within a frame. Unnecessary transmission and maintenance of useless packets in the buffers cause further loss and degrades the quality of delivery (QoD) significantly. In this paper, we propose a QoD-aware hop system that can decide when and which packets could be dropped without degrading QoD. Moreover, the transmission of useless packets causes network congestion and vain payment by the wireless system subscriber. In this paper, we focus on two types of frame discarding policies to maintain QoD; partial frame discarding policy (PFD) and early frame discarding policy (EFD). PFD policy discards the succeeding packets of a frame if a packet of the frame cannot be served. On the other hand, in EFD policy when it is likely to fail to serve packets of a frame (based on a threshold), the subsequent packets of the frame are discarded. We first provide an analytical study of average buffer occupancy based on these discarding policies and show the closed-form expressions for average buffer occupancy. We then perform our simulations by implementing a Markovian model and measure the frameput (the ratio of number of frames served) rather than the number of packets served.

KEYWORDS: Average buffer occupancy, discarding policies, traffic reduction.

1 INTRODUCTION

Multimedia communication over wireless networks has gained popularity in the last five years. However, the bandwidth limitation, low quality video transmission, synchronization, and price cost endured by subscribers are some of the major drawbacks of wireless networks. On the other hand, one of the major advantages of wireless networks is the mobility characteristic which refers to the easiness of moving around while staying connected to the internet [32]. However, the transmission medium over wireless networks is the open air, which is unlike the fiber optical backbone and copper cable found over wired networks [33] that have the consequences of appearing new uncontrollable factors such as multipath interference, weather conditions, and urban obstacles [33]. This makes the bit error rate (BER) in wireless networks to be much higher than in wired networks [32][33]. The frequent handoffs that result temporal disconnections between communicating end hosts due to the limitations of radio coverage and user mobility is another drawback of wireless networks [33]. One more consequence of adapting wireless network is the link (or rate) asymmetry that refers to the situation where the forward and reverse paths of a transmission have different channel capacities [32][33][34]. Actually, the wireless hops are likely to be congested due to asymmetric rate of arrival vs. outgoing packets. At this point, it is very hard for wireless networks to reach the bandwidth supported by wired networks. As long as the asymmetry exists for the wireless routers, the network congestion and dropping packets are inevitable in the case of demand for multimedia data [22]. Since real-time video streaming usually uses User Datagram Protocol, the retransmission of packets is not an option to improve the quality of delivery (QoD). We define the QoD as the best effort strategy to increase the integrity of service using available bandwidth without promising or pre-allocating of resources for the sender (i.e., traffic contract) as in quality of service (QoS). The major goal in QoD is to maximize the quality under given specific resources without any dedication for the sender. Therefore, our strategies enhance the quality of service (or quality of data) obtained at the receiver. Dropping packets randomly (due to overflows) degrades the QoD significantly. In this paper, we try to reduce the network congestion by providing a QoD-aware hop system that accounts for the dependencies among frames in videos. Moreover, since garbage packets cannot be used at the receiver side; the QoD-aware hop system does not retransmit these packets and helps save money for the subscribers.

The wireless local area networks are created by wireless access points (routers) that are wired to a local area network. If the discrepancy of rates between the incoming and outgoing network is high, most of the packets are dropped due to buffer overflows [24]. Trad et al try to adjust the rate of VoIP to overcome this problem [34]. The frames are unnecessarily dropped until the sender reacts to the congestion. Some of the packets that are received by the client cannot be decoded due to the loss of (dependent) packets that are required for decoding. This is worse for mobile customers who pay for the number of bytes transmitted. If the transmitted packets are useless in decoding a complete frame, the mobile customer pays for useless transmission.

There are a lot of ideas which have been proposed to enhance the performance of multimedia streaming over wireless networks. One of the good approaches was to combine the well-known techniques of dynamic resource selection and dynamic content adaptation to resolve inherent problems of multimedia streaming regarding congestion and interference over wireless networks [27]. This was based on the use of unified link layer API not to only tailor the video transmission with respect to the wireless link performance, but also to configure the links to react to the environmental changes or performance obstacles [27]. Another method on multimedia streaming over wireless networks incorporated unequal error protection (UEP) coding technique with multimedia applications for better utilization of channel resources [28]. On the other hand, a study was conducted for maximizing throughput and minimizing delay of multimedia streaming over infrastructureless networks by proposing a greedy method based on directed diffusion that seeks for routes' reinforcement for high link quality and low latency [29]. To measure the link quality, the expected transmission count (ETX) metric was used [29].

Message discarding policies have been proposed [3] [4] for TCP/IP-based transmission for messages and email files to reduce network congestion. Consequently, it has been proven that message-based discarding policy mechanism provides a remarkable improvement in network performance compared to systems with no control policy. Basically the TCP/IP-based system [2] [5] is an example of a message that is segmented into packets that need to be transmitted and then reassembled back again at the receiving end. There was also a suggestion for implementing these discarding policies for ATM networks that use AAL5 as an adaptation layer to the ATM layer [10]. However, previous research has been done using these message or email files discarding policies by providing numerical [5] [6] study, analytical (using generating functions) study ([7] for Partial Message Discarding policy) and [8] for Early Message Discarding Policy (EMD)), fluid study [7] [19] [20] [21], and analytical study using discrete-time [18] to find the goodput and queue-length distribution. In this paper, we use the same continuous-time analytical method (z-transform) as previous research, but to find closed form expressions for the average buffer occupancy. For details about this method, see [7] [8]. On the other hand, it has been proposed a discrete-time queue for the Early Message Discarding policy in high-speed networks considering bursty arrival and server interruptions (i.e., temporary server unavailability caused by sharing a server with other buffers) [30]. The analysis has been made using quasi-birth-and-death process to derive the steady-state probability distribution of buffer content using probability generating-functions approach [30]. There are drawbacks of this approach: arriving packets are generated according to Bernoulli process. In other words, only one packet can arrive in any time slot. Hence, the maximum number of arriving packets can be determined which is the maximum adapted simulation time. In reality, this is not true since we cannot predict about the incoming load especially under high-speed networks. Furthermore, within a time slot, at most one packet can get served since the service times follow geometric distribution. Therefore, under multimedia communication a continuous-time queue is of interest due to high arriving and departing packet rates.

Since we use frames in video streaming, these discarding policies are named as partial frame discarding (PFD) and early frame discarding (EFD) policy in our case. We should also note that we consider User Datagram Protocol (UDP) for smooth video presentations instead of TCP. In PFD, if a packet of a frame is lost, the rest of the packets belonging to the same frame are dropped, because a frame cannot be decoded properly with loss of packets. In EFD policy, when it is likely that the buffer will not be able to handle future packets of a frame, those future packets are dropped. Moreover, the acceptance of new frames is controlled by a threshold. Thus, EFD policy does not only reject the rest (tail) of the frames as PFD, but also rejects complete frames. In our earlier work, we have provided an analytical study of average buffer occupancy and loss probability [26]. We have simulated the performance of discarding policies and analyzed the packet loss and goodput with respect to arrival rate [25].

Our Approach. How can we maintain QoD for real-time video streaming in wireless networks having high traffic conditions? Under high traffic conditions, the QoD cannot be achieved by using traditional policies (with no control). We achieve QoD by using three mechanisms: a) use of QoD-aware policy for buffering, b) increase of buffer without increasing delays for packets, and c) sending a jamming signal (peer-to-peer acknowledgment) that is quick and short to the sender to reduce the frame rate (slow down). For buffering, to maintain QoD we study, analyze, and simulate two types of discarding policies: partial frame discarding (PFD) and early frame discarding (EFD) policies. Even these discarding policies might not be satisfactory to maintain QoD if the buffer size is not chosen properly. Determining the average buffer occupancy for different traffic conditions and frame (packet) sizes improves the QoD significantly. If increasing buffer size is still not satisfactory, a jamming peer-to-peer signal is sent from the hop to the sender to reduce the frame rate for the video.

In this paper, we analyze and simulate an intermediate QoD-aware hop (router) system with finite buffer using frame discarding policies to get best buffer utilization by neglecting packets belonging to corrupted frames. This model is considered to be working at the link layer of intermediate hops systems. In our analytical study, we evaluate the performance of an intermediate QoD-aware hop system by utilizing frame discarding policies and providing an explicit expression for the average buffer occupancy. We have analyzed and measured the performance of two types of discarding policies at the QoD-aware hop system: partial frame discarding (PFD) policy and early frame discarding (EFD) policy. In our analytical and simulation studies, we also include a (no-control) system without including any type of traffic control.

In this paper, we also theoretically analyze and extract the (closed form) expressions for the average buffer occupancy for all discarding policies. Since we use a finite buffer, it is critical to know the average buffer size. The determination of average buffer occupancy is important since unsatisfactory buffer size may cause overflows even when the best discarding policies are selected. Small buffer size increases the loss probability of packets. In the same way, the use of huge buffer size corresponds to under-utilization of system resources. Choosing a large buffer size also leads to a long waiting time which is not acceptable for real-time streaming applications. Therefore, inappropriate buffer size would worsen QoD even when using these discarding polices.

Besides the theoretical analysis, we have also performed simulation studies using the Markovian model for each discarding policy separately. We have used the memoryless M/M/1/N model using continuous-time Markov model. Basically M/M/1/N refers to negative exponential arrivals and service times of packets (i.e. generated by a Poisson process) with a single server and a finite buffer size (N). The frame size is considered to be geometrically distributed with parameter q (i.e. the mean size of a frame as $1/q$ packets).

In our simulations, we have measured an important QoD performance metric termed as frameput at the frame level. We define it as the accumulative number of good frames being served to the accumulative number of arriving frames. Through this frame level performance metric we can have an informative picture about these discarding policies when applied on intermediate QoD-aware hop system (like router). Moreover, we simulate an intermediate QoD-aware hop system without any control policy (this is the case we may find nowadays for video over Internet Protocol (IP) transmission). We explain through detailed flowcharts how to simulate the intermediate QoD-aware hop system by applying these discarding policies. We show and compare the results for these performance metrics for all discarding policies for different traffic conditions (i.e. different mean arrival rates and mean service times) and for different mean frame lengths (in terms of packets). The simulation study is not a verification of analytical results by simulations. The simulation study further performs experiments how frame discarding policies affect the quality or frameput. The analytical study helps us determine proper resource allocation while the simulation studies show how quality is affected for a given buffer size.

Our contributions can be listed briefly as follows:

- Employment of discarding policies for wireless multimedia communications
- Providing the closed-form expressions for average buffer occupancy for EFD, PFD, and no control policies
- Simulation of a Markovian model by incorporating discarding policies and determining the best policy for wireless multimedia communications
- Reduction of network congestion by using a reasonable buffer size
- Increasing QoD for wireless customers
- Reduction of cost for wireless customers.

This paper is organized as follows. The following section provides background on video compression, MPEG video compression and video transmission. Section 3 discusses the discarding policies. Section 4 provides the theoretical analysis and closed-form expressions for the average buffer occupancy. Section 5 explains the simulation setup and simulation results. The last section concludes the paper.

2 BACKGROUND ON VIDEO COMPRESSION AND STREAMING

This section includes two subsections. Subsection 2.1 discusses video compression. Subsection 2.2 illustrates video streaming.

2.1 Video Compression

Video compression exploits two types of redundancy: spatial redundancy and temporal redundancy. The temporal redundancy is minimized by using motion compensation methods. The spatial redundancy removal is achieved by compression methods like Discrete Cosine Transform (DCT) and Discrete Wavelet Transform. In this paper, for real-rime video streaming we study MPEG video streaming. Actually, our method is applicable to any video format that uses dependencies among blocks in frames as well as dependencies among frames. Since there is compression for videos, the delivery of good frames as much as possible is very important to achieve a best QoD.

The temporal and spatial redundancy elimination among frames introduced 3 types of frames in MPEG video: *I*, *P*, and *B* frames. I frames can be decoded independent of other frames and is called intra-frames but it only minimizes spatial redundancy. MPEG-1, MPEG-2, H.264, and MPEG-4 use DCT to remove the spatial redundancy within a frame. P-frames are predicted from the previous P-frame or I-frame. B-frames are generated by interpolation of the previous P-frame (or I-frame) and the successive P-frame (or I-frame). The MPEG video is generated using IPB patterns. IBBPBBPBBPBBPBB is one of the most common patterns that are used in MPEG encoding. Since the expected playback rate for videos is at least 30 frames per second (fps) nowadays, this type of pattern helps the player recover within half a second in case of errors. Each pattern corresponds to a Group of Pictures (GOPs) that start with I-frames. The loss of I-frame in a GOP results in loss of all the other frames in the GOP. The loss of P frames results in the loss of all the successive P-frames and B frames. In terms of sizes of frames, the size of I-frames is more than the size of a P-frame. The size of a P-frame is more than the size of a B-frame.

Besides, the dependency among frames, there is also a dependency within each frame. Each frame is divided into slices in MPEG coding. The slices are the smallest unit that can be decoded independently. The slices are composed of macroblocks. The macroblocks are composed of 6 blocks: 4 for *Y* (luminance) components and 2 for *Cb* and *Cr* (color) components in 4:2:0 color sampling. The DC coefficients of blocks are compressed using Differential Pulse Code Modulation (DPCM). The loss of the first macroblock in a slice results in the loss of the successive macroblocks in the same slice.

2.2 Video Streaming

Multimedia applications use UDP for real-time video streaming. In this paper, we apply the discarding policies for video transmission where there is no retransmission in the case of dropping frames packets. Since videos are compressed and there is a dependency within a frame as well as among frames, the loss of packets or frames may degrade the QoD significantly.

The major concern in wireless multimedia networks is the transmission of video data with high quality. MPEG video format [15] is one of the widely used formats in digital video communications. In [13], the sample mean frame sizes are listed as 197.1 Kbits, 58.0 Kbits, and 19.6 Kbits for I-frame, P-frame, B-frame, respectively. In [12], the typical packet size for video data using User Datagram Protocol (UDP) is mentioned as 200 bytes. In [14], the packet size for video is considered as 1024 bytes. The number of packets typically ranges from 24 to 123 for I-frames, from 7 to 36 for P-frames, and from 2 to 13 frames for B-frames based on these different packet sizes. We have considered the number of packets in these ranges in our analysis. Without loss of generality, from the perspective of a single frame, we may assume that the loss of a block may deteriorate the quality of the frame, and the whole frame may be considered as lost. This assumption is valid since incorrect blocks distract the watcher and get the attention of the user to the incorrect blocks. The same idea may also apply to a group of pictures (GOPs). When one frame is lossy, the rest of the frames cannot be recovered especially when only I and P frames are used since each P frame depends on the previous frame. H.261 video format [16] uses only I and P frames. There is also research for new standards like H.264 that targets only IPPP patterns [35]. In this paper, we are going to use the term frame as the smallest meaningful data that can be interpreted by the user.

3 NETWORK AND BUFFER MODEL FOR QOD-AWARE HOP SYSTEM

In our analysis, we use the same network and buffer model covered in [1] and [5]. However, we apply their methodology on messages to video data. In real-time streaming applications, video streaming is usually accomplished using UDP and the control messages are transmitted by using TCP [17]. A frame in the application layer is fragmented into packets and transmitted as a series of packets. The decoding at the receiver side starts after all the packets of a frame are received and assembled. As mentioned in Section 2, the frames may have varying sizes. In this paper, we have similar assumptions on packet arrival rate as in [5]. In [5], packets arrive with a geometrical distribution as a parameter of q and this corresponds to the mean size of a frame as $1/q$ packets.

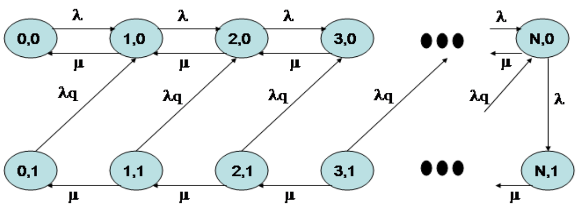

Figure 1 State transitions for PMD Policy [5] [7]

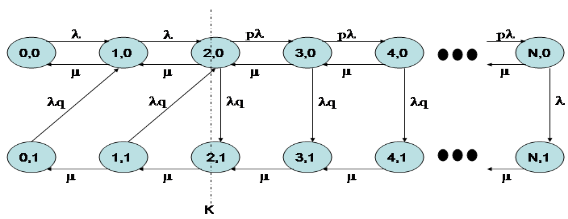

Figure 2 State transitions for EMD Policy [5] [8]

In our system, a buffer can maintain at most N packets. If a packet arrives when there are N packets in the buffer, that packet is discarded and considered as lost. The packets are removed from the buffer after the packets of a frame are assembled and decoded. In [5], the network is modeled according to the M/M/1/N model, with arrival rate λ , service rate μ , and the load on the network element as $\rho = \lambda / \mu$.

We have employed two types of discarding policies: PFD and EFD. In PFD policy, if a packet of a frame arrives at a full buffer, then this packet and all consecutive packets that belong to the same frame are discarded [5]. There are two modes in this policy: normal and discarding. In the normal mode, when a packet arrives, it is stored in the buffer. In the discarding mode, a packet is discarded due to the full buffer or discarding a packet of the same frame in the past. The PFD policy diagram is depicted in Figure 1 [5], [7]. The top side of Figure 1 corresponds to the normal mode and the bottom side represents the transitions for the discarding mode. A state is represented as a pair (*i*,*j*) where *i* represents the number of packets in the buffer and *j* represents the mode (0 for normal and 1 for discarding). Since the length of a frame is geometrically distributed, the probability of a packet belonging to the same frame is $(1 - q)$; and this probability is also equivalent to the probability to discard a packet [5]. We call the first packet of a frame as a head-of-frame packet. A head-of-frame packet arrives with probability q [5]. If $P_{i,j}$ $(0 \le i \le N$, $j = 0, 1)$ be the steady-state probability of having *i* packets in the system with the system in mode *j*, the following set of equations are derived in [5]:

$$\rho P_{0,0} = P_{1,0}, \tag{1}$$

$$q\rho P_{0,1} = P_{1,1}, \tag{2}$$

$$(\rho + 1)P_{i,0} = \rho P_{i-1,0} + P_{i+1,0} + q\rho P_{i-1,1}, \quad \text{for} \quad 1 \le i \le N-1, \tag{3}$$

$$(q\rho + 1)P_{i,1} = P_{i+1,1}, \quad \text{for} \quad 1 \le i \le N-1, \tag{4}$$

$$(\rho + 1)P_{N,0} = \rho P_{N-1,0} + q\rho P_{N-1,1}, \tag{5}$$

$$P_{N,1} = \rho P_{N,0}. \tag{6}$$

EFD policy employs a threshold *K* to maintain the buffer occupancy where K is an integer and $0 \le K \le N$ [5]. If a frame starts to arrive when the buffer occupancy is at or above K packets, then all the packets of that frame are discarded. The threshold *K* must be chosen carefully. If *K* is chosen to be too low, the buffer is not well utilized since many frames that may have been accepted are discarded [5]. Similarly, if *K* is too high, the system acts almost like a PFD policy and loses its main and relative advantage [5]. The state transition diagram for EFD policy is given in Figure 2 [5] where (*p*=1-*q*).

If $P_{i,j}$ $(0 \le i \le N$, $j = 0, 1)$ is the steady-state probability of having *i* packets in the system with the system in mode *j*, the following set of equations is derived in [5]:

$$\rho P_{0,0} = P_{1,0}, \tag{7}$$

$$q\rho P_{0,1} = P_{1,1}, \tag{8}$$

$$(\rho + 1)P_{i,0} = \rho P_{i-1,0} + P_{i+1,0} + q\rho P_{i-1,1}, \quad \text{for} \quad 1 \le i \le K \tag{9}$$

$$(\rho + 1)P_{i,0} = (1-q)\rho P_{i-1,0} + P_{i+1,0}, \quad \text{for} \quad K+1 \le i \le N-1, \tag{10}$$

$$(\rho + 1)P_{N,0} = (1-q)\rho P_{N-1,0}, \tag{11}$$

$$P_{N,1} = \rho P_{N,0}, \tag{12}$$

$$(q\rho + 1)P_{i,1} = P_{i+1,1}, \quad \text{for} \quad 1 \le i \le K-1, \tag{13}$$

$$P_{i,1} = P_{i+1,1} + q\rho P_{i,0}, \quad \text{for} \quad K \le i \le N-1. \tag{14}$$

4 THEORETICAL ANALYSIS

This section contains two subsections. Subsection 4.1 shows the theoretical analysis and the explicit expressions for the average buffer occupancy for all frame discarding policies. Subsection 4.2 illustrates the analytical results.

4.1 Analysis and Closed Form Expressions

For both policies, we have:

$$\sum_{i=0}^{N} (P_{i,0} + P_{i,1}) = 1. \tag{15}$$

For both policies, the following generating function is defined:

$$P_j(z) = \sum_{i=0}^{N} P_{i,j}\, z^i. \tag{16}$$

And the generating function for the number of packets in the queue is

$$P(z) = \sum_{j=0}^{1} P_j(z). \tag{17}$$

And the average number of packets in the buffer is the derivative of P(z) at z =1

$$\frac{d}{dz} P(z)\Big|_{z=1} = \sum_{j=0}^{1} \sum_{i=0}^{N} iP_{i,j}. \tag{18}$$

$$\frac{d}{dz} P(z)\Big|_{z=1} = E[X],$$

where X is the number of packets in the system.

Whenever the derivation is like $\frac{d}{dz} P(z)\Big|_{z=1} = 0/0$ then we must take the limit and use L'Hopitals rule. For information about *p(z)* see [7][8].

Average buffer occupancy (ABO) for EFD policy can be expressed by the following summation:

$$ABO = \sum_{i=1}^{4} G_i, \tag{19}$$

where G_1 is defined as

$$G_1 = \frac{B_1 B_2 - [B_3 B_4 + B_5 B_6]}{2(B_2)^2}, \tag{20}$$

$$B_1 = f_6\rho\big(2 - 2(k+1) - q\rho\, k(k+1)\big) + 2f_5(1 - q\rho) + B_a, \tag{21}$$

$$B_a = [(1 - f_1)f_6]\big((k-1)(2 + q\rho\, k)\big) + \big(f_6(\rho + f_1 - 1)\big)k(k-1)q\rho, \tag{22}$$

$$B_2 = (q\rho)(1 - \rho), \tag{23}$$

$$B_3 = q\rho\,\big(f_6(\rho + f_1 - 1)\big) + (q\rho)f_6(1 - f_1 - \rho), \tag{24}$$

$$B_4 = 6q\rho - 6(1 - \rho) - 6, \tag{25}$$

$$B_5 = 2 - 2\rho(1 + q), \tag{26}$$

$$B_6 = q\rho\, f_5 - f_6\rho(q\rho(k+1) + 1) + (1 - f_1)f_6\,(q\rho\, k + 1) + B_b, \tag{27}$$

$$B_b = q\rho\left(k\left(f_6(\rho + f_1 - 1)\right) + \frac{f_6(\rho + f_1 - 1)}{\rho q f_2}\right), \tag{28}$$

$$f_6 = \frac{f_2}{f_1 r\left[\frac{\rho^{2-k} f_2}{(1-\rho)(1-r\rho)} + \frac{1}{(1-r)(1-r\rho)}\right] + \frac{f_2 W}{(1-\rho)(1-r)\rho}}, \tag{29}$$

$$W = (1 - f_1)(\rho - 1 + Kq\rho^2) + \frac{1}{f_2}(\rho + f_1 - 1) - W1, \tag{30}$$

$$W_1 = \rho^2(2 - r + Kq\rho) + \frac{f_4}{r\rho f_3}(1-\rho)\rho^3 r^2 - W_2, \tag{31}$$

$$W_2 = \rho(\rho + f_1 - 1)(1 - \rho + Kq\rho), \tag{32}$$

$$f_1 = \frac{z_1^{K-N}\left(\frac{1}{z_1} - r\rho\right) + z_2^{K-N}\left(r\rho - \frac{1}{z_2}\right)}{f_3}, \tag{33}$$

$$z_{1,2} = \left((\rho + 1) \pm (\rho(\rho - 4r + 2) + 1)^{1/2}\right) / 2r\rho, \tag{34}$$

$$f_2 = \sum_{m=0}^{k-1} \binom{k-1}{m}\left[(1-r)\rho\right]^{k-m-1}, \tag{35}$$

$$f_3 = z_1^{K-N-1} - z_2^{K-N-1}, \tag{36}$$

$$f_4 = -\left(\frac{1}{r\rho}\right)^{K-N-1}\left(\rho(\rho - 4r + 2) + 1\right)^{1/2}, \tag{37}$$

$$f_5 = \frac{f_1 f_6 r}{1 - r\rho}\left(\rho^{2-k} + \frac{1}{qf_2}\right), \tag{38}$$

G_2 is defined as

$$G_2 = \frac{C_1 C_2 - (C_3 C_4 + C_5 C_6)}{2(C_2)^2}, \tag{39}$$

$$\begin{aligned} C_1 &= \left(f_6(\rho + f_1 - 1)\right)\left[2 + 2k(r\rho - 1) + k(k-1)\rho(r-1)\right] \\ &\quad - \left((1 - f_1)f_6\right)kq\rho(k-1) + C_a, \end{aligned} \tag{40}$$

$$C_a = f_6 q\rho\left(2k - 2 + \rho k(k-1)\right) + \left(\frac{f_6 f_4}{f_3}\right)2r\rho(N(1-\rho) - 1), \tag{41}$$

$$C_2 = \rho(1 - r), \tag{42}$$

$$C_3 = \left(f_6(\rho + f_1 - 1)\right)\rho(r-1) - f_6 q\rho\left(1 - f_1 - \rho\right), \tag{43}$$

$$C_4 = 6\rho(1 - r) - 6(1 - r\rho) - 6, \tag{44}$$

$$C_5 = 2(1 - r\rho) - 2\rho(1 - r), \tag{45}$$

$$C_6 = \left(f_6(\rho + f_1 - 1)\right)\left((r\rho - 1) + k\rho(r-1)\right) - (1 - f_1)f_6 kq\rho + C_b, \tag{46}$$

$$C_b = q\rho f_6(1 + k\rho) + \left(\frac{f_6 \quad f_4}{f_3}\right)\rho(r(1-\rho)). \tag{47}$$

and G_3 and G_4 are defined as

$$G_3 = \frac{(\rho q + 1)(k - \frac{1}{q\rho})(f_6(\rho + f_1 - 1)\ ) + \frac{f_6(\rho + f_1 - 1)}{\rho q f_2}}{q\rho}. \tag{48}$$

$$G_4 = \frac{r\rho f_6\left(q\rho(k+1) - (1 - r\rho)\right) - W_3 - W_4}{(q\rho)^2}, \tag{49}$$

$$W_3 = ((1 - f_1)f_6)(q\rho k - (1 - r\rho)), \quad r = 1 - q, \tag{50}$$

$$W_4 = \left(\frac{f_6 \quad f_4}{f_3}\right)r\rho\left(q\rho(N+1) - (1 - r\rho)\right). \tag{51}$$

Average buffer occupancy (ABO) for PFD policy can be derived as the following summation:

$$ABO = \sum_{i=1}^{2} F_i, \tag{52}$$

where**,**

$$F_1 = \frac{\left((1 - t \times t_1) + q\rho(N+1)t\ \right)\left[\rho(r(1-\rho)+1) - 1\right]}{\rho(1-r)Q_1}, \tag{53}$$

$$t = (1 + (1-r)\rho)^N \quad \text{and} \quad t_1 = 1 + (1-r)\rho, \tag{54}$$

$$Q_1 = \rho(r(1 - t\rho) + t) - 1 + (1-r)(1 - \rho^{-N}t), \tag{55}$$

$$F_2 = \sum_{j=1}^{N} \frac{j \times t_1^{\,j-1} Q_2 \left[1 - \left(\frac{t_1}{\rho}\right)^{N-j+1}\right]}{(\rho - t_1)Q_1}, \tag{56}$$

$$Q_2 = r^2\rho^2(\rho - 1) - \rho^2(r\rho - 1) - q\rho. \tag{57}$$

Average buffer occupancy for no control policy can be derived as:

$$ABO = \frac{\rho\left(1 - y\left[N\left(1 - \rho q^{-1}\right) + 1\right]\right)}{q - \rho\left(y\left(1 - \rho q^{-1}\right) + 1\right)}, \tag{58}$$

where

$$y = \left(\rho/q\right)^N. \tag{59}$$

4.2 Analytical Results

We start with how the buffer size affects the ABO under the same traffic conditions. We provide results for PFD to see the effect of different system capacities. Rather than showing the results using the same parameters, we use different system capacities for PFD to show the effect of different buffer sizes and to save space. Figure 3 represents the average buffer occupancy versus arrival rate for PFD policy for different system capacities. For a fixed mean frame length (20 packets), service rate (0.2), and system capacity (70 or 80), the increase in the average buffer occupancy as the arrival rate increases is noticed. This is expected since the traffic (system flow) increases as the arrival rate increases. It is seen that the curve structure is almost the same for different buffer sizes. Furthermore, it is also very important to notice the importance of finding an explicit expression for ABO. If the traffic load increases, the packets start arriving to a full buffer. Thus, this degrades the QoD since it indicates there is an increase on dropping (discarding) packets. Therefore, this theoretical analysis gives us a great picture for the necessity of increasing the buffer size for those high traffic conditions. Further, it is also important to choose the necessary (or required) buffer size when the traffic load goes down. For example, choosing the buffer size as 70 leads to a long waiting time. Consequently, a lower frame rate for playback at the receiver side is achievable. Hence, the QoD is getting even worse and the customers pay for that undesired QoD which is absolutely not the target.

Figure 4 illustrates the average buffer occupancy versus arrival rate when applying PFD policy. It can be seen from Figure 4 that for a fixed system capacity and a fixed traffic load (λ / μ) , the average buffer occupancy (ABO) increases as the mean frame length decreases. This is due to the advantage of using PFD policy. We know that the main purpose of PFD policy is to reject the tail packets of a frame. Therefore, increasing ($1/q$) means increasing the frame size in terms of packets. This also corresponds to increase the chance to reject more and more tail packets. Consequently, the ABO is getting lower when the mean frame size is increased. We can think of (q) as another explanation for this trend. Hence, when the mean frame length ($1/q$) decreases, this means there is a high probability for any head-of-frame packet to arrive to the intermediate QoD-aware hop system. Consequently, this leads to an increase in the average buffer occupancy (ABO). For a fixed mean frame length, service rate, and system capacity, the increase in the average buffer occupancy as the arrival rate increases is noticed.

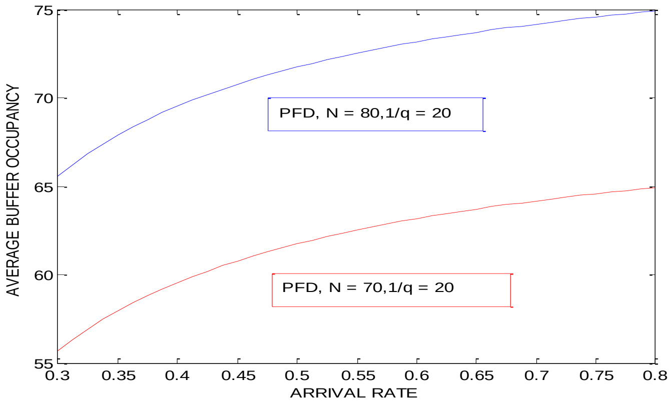

Figure 3 Average buffer occupancy versus arrival rate when applying PFD policy (service rate = 0.2; N = 70, 80; $1/q$ = 20)

Figures 5 and 6 present the average buffer occupancy versus arrival rate when applying the EFD policy. It is shown that for a fixed system capacity and threshold buffer occupancy level, the ABO increases as mean frame length decreases. It is important to consider that the purpose (advantage) of EFD policy over the QoD-aware hop system is not only to reject tail packets that belong to the same frame as PFD but also to reject complete frames. The ABO increases as the arrival rate increases for a fixed system capacity, threshold buffer occupancy level, and mean frame length. It is also shown in Figures 5 and 6 that for a fixed system capacity and mean frame length, the increase of ABO as the threshold buffer occupancy level increases. This is expected since there is high traffic (load>1).

It is also interesting to see from Figure 7 that the average buffer occupancy reaches the maximum system capacity under traffic load (λ / μ) much lower than in PFD and EFD. Actually, this trend of shape clearly indicates there is a high probability of dropping packets. Therefore, the customers are unfortunately served low QoD. Hence, the buffer size should be increased to avoid such a situation. That is why finding closed form expression is so important. It is also interesting to see from this figure that ABO increases as the mean frame length increases and this is not the case for PFD and EFD policies. This is expected since in a no control policy there is no subsequent packet rejection at all. Thus, increasing the mean frame size with no rejection policy means getting an increase in the ABO.

In comparison, it is seen from Figures 4, 5, 6, and 7 that the ABO in PFD is lower than NO CONTROL. It is also noted that ABO in EFD is lower than PFD.

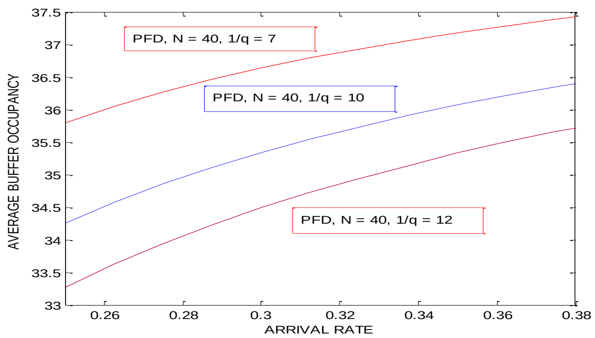

Figure 4 Average buffer occupancy versus arrival rate when applying PFD policy (service rate = 0.13; N = 40; $1/q$ = 7, 10, and 12)

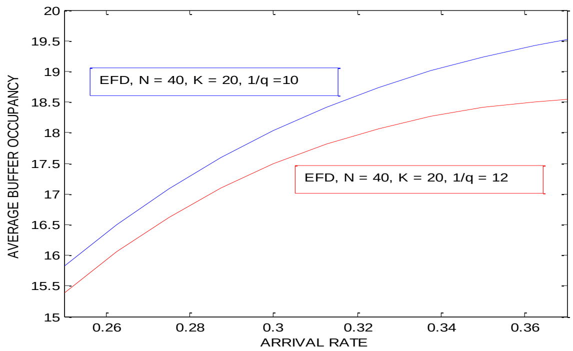

Figure 5 Average buffer occupancy versus arrival rate when applying EFD policy (service rate = 0.13; N = 40; K = 20; $1/q$ = 10, 12)

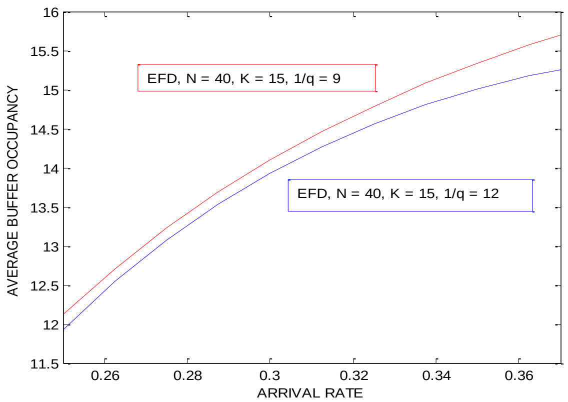

Figure 6 Average buffer occupancy versus arrival rate when applying EFD policy (service rate = 0.13; N = 40; K = 15; $1/q$ = 9, 12)

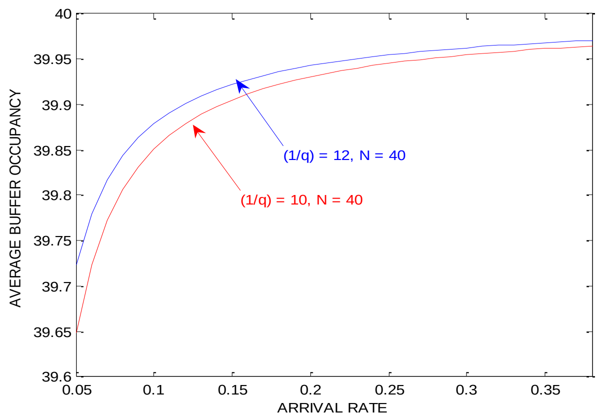

Figure 7 Average buffer occupancy versus arrival rate when applying NO CONTROL policy (service rate = 0.13; N = 40)

5 SIMULATION ENVIRONMENTS

In this section, we describe the simulation environment for all frame discarding policies. There are three subsections. Subsection 5.1 shows the simulations of the used distributions. Subsection 5.2 illustrates the simulation setup and flowcharts of our simulators (for each policy). Subsection 5.3 discusses the simulation results for the frameput performance metric. We provide the simulations with respect to within frame dependencies but it can be easily extended dependency within a GOP (especially for IPPP… patterns).

5.1 Simulation of Distributions

The Geometric distribution is a discrete and memoryless distribution. We can generate Geometric random numbers in MATLAB by using the inverse CDF (cumulative distribution function) [23] method like the following:

- Generate u ~ unif (0, 1), where u is uniformly distributed random number the region of [0,1].
- Compute $Y = F_X^{-1}(u)$, where $F_X^{-1}(x)$ is the inverse CDF of the Geometric distribution.

$$CDF = P(X \leq k) = 1 - (1 - q)^k \text{, where } k = 0,1,2,3,.. \quad . \tag{60}$$

$$\text{Thus, } Y = 1 + floor\left\lfloor \frac{\ln(U)}{\ln(1 - q)} \right\rfloor. \tag{61}$$

The Poisson distribution (discrete distribution) is simply the Poisson process which is a birth process in stochastic modeling. We can generate Poisson random numbers in MATLAB simply by using the following function:

Arrival_num = random ('*Poisson'*, *Arrival_rate*).

5.2 Simulation Setup and Flowcharts

The simulations of EFD, PFD, and NO CONTROL policies are performed using MATLAB. We measure the QoD performance of our intermediate QoD-aware hop system with frameput metric. Figure 8 shows the flowchart of the program and demonstrates the major steps to implement the intermediate QoD-aware hop system with initial conditions. The upper side of Figure 8 shows the initializations for the simulation. The left-bottom side of Figure 8 keeps statistics about the simulation whereas the right-bottom side manages which policy to use. The middle-bottom of Figure 8 generates the arrival of packets with aforementioned probability distributions.

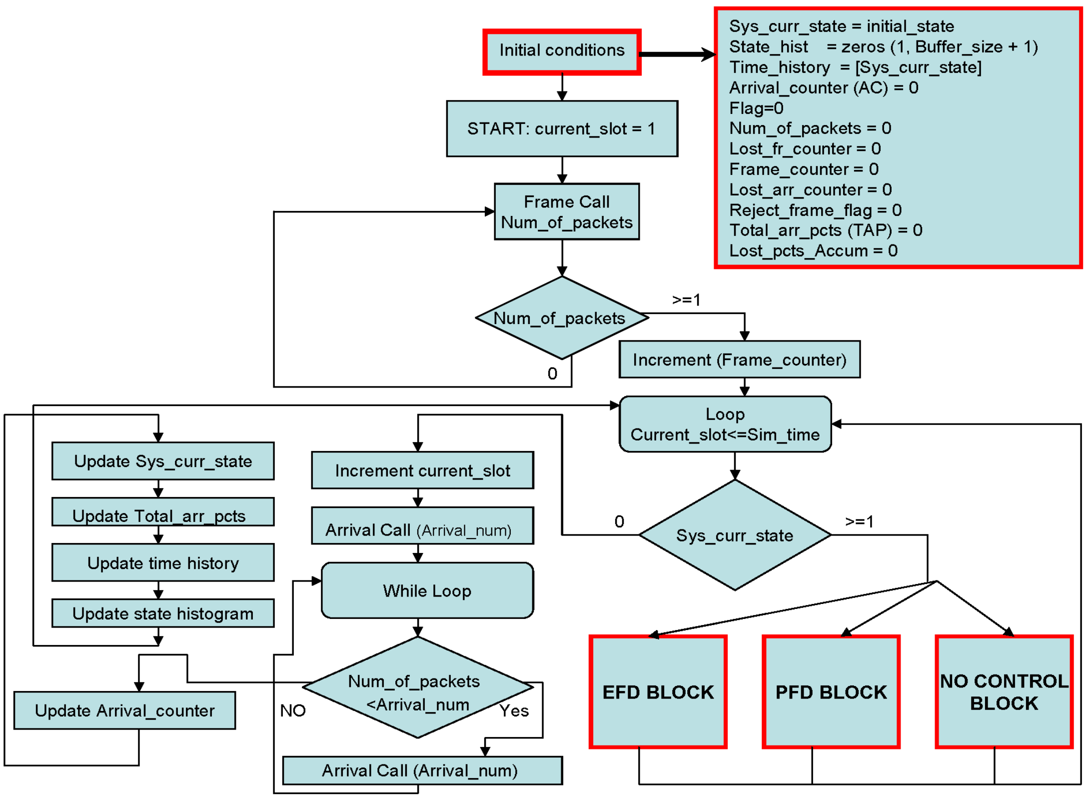

Figure 8 Major steps and initial conditions to implement the intermediate QoD-aware hop system

Figures 9, 10, 11, 12, and 13 show the common steps in all policies. Figure 9 displays the steps when serving a frame is completed. Figure 10 shows the general steps for service. Figure 11 shows some of the steps taken when there is a packet loss. Figure 12 shows steps for arrival packets. Figure 13 displays steps for updating statistics.

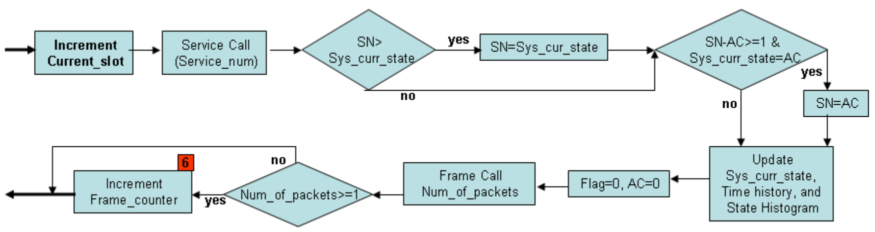

Figure 9 Simulation steps for processing a frame (Process Frame Block)

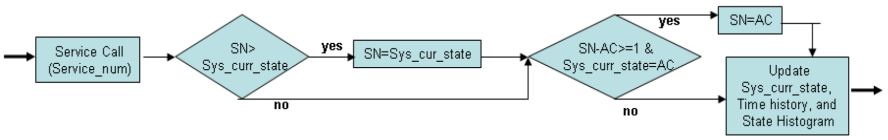

Figure 10 Simulation steps to serve packets (Serve Packet Block)

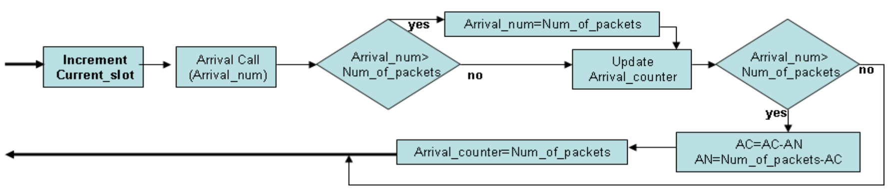

Figure 11 Partial simulation steps taken when a packet of a frame is lost (Process Incomplete Frame Block)

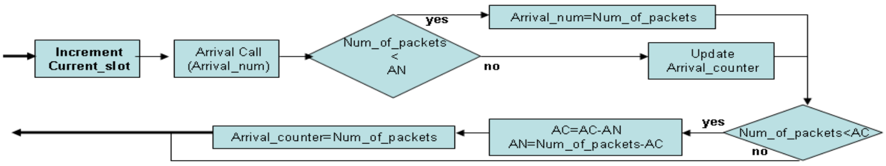

Figure 12 Partial simulation steps to receive packets (Process Packet Block)

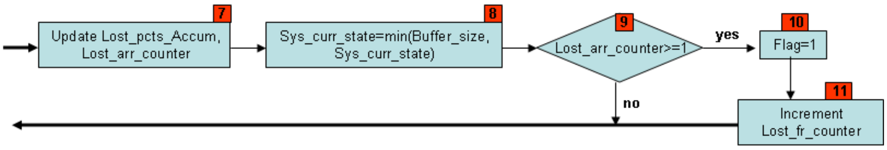

Figure 13 Steps taken to update statistics (Update Statistics Block)

Figure 14 shows the major steps to implement the NO CONTROL policy. Figure 15 presents the major steps to implement the PFD policy. Figure 11 shows the major steps to implement a system with EFD policy.

It is seen from Figures 8, 9, 10, 11, and 12 there are three cases of using random number generators. The first case occurs when calling Geometric random number generator (RNG). The input to this RNG is q, which is the probability for a head-of-frame's packet to come. The output of this RNG is the number of packets in a certain frame.

The second case occurs when calling Poisson RNG. The input to this RNG is arrival rate (λ). The output of this RNG is the number of arriving packets. Absolutely, this number should refer to a piece (or a whole) of frame.

One of the critical issues in the simulation of all discarding policies (Figures 8, 11, and 12) is the synchronization. Since the outputs from Poisson and Geometric number generators are random numbers, there must be a kind of strong synchronization through a good connection between generated frame’s packets and arriving packets to the QoD-aware hop system. For example, it is necessary to make sure that the number of arriving packets refers to a specific frame (size) with a non-zero value. It is important to make sure that the arriving packets must be less than or equal to the total packets of a certain frame. Further, we must make sure that the accumulative number of arriving packets must not exceed the total number of packets for a specific frame.

The third case occurs when calling Poisson RNG to express or represent the number of packets being served at a certain slot time. The input to this RNG is the service rate (μ). The output of RNG is the number of packets being served. Definitely, this number should refer to the number of arriving packets and system buffer. Hence, a connection between the service mechanism and the number of packets waiting for service at the system buffer must be considered. For example, it is meaningless to issue service for an empty buffer having no packets waiting for service. Furthermore, it is meaningless to have a random number of serving packets greater than the number of packets currently waiting in the buffer. Thus, a meaningful connection (or synchronization) between the outputs of Poisson (arriving and serving) and Geometric (frame size in terms of packets) has to be taken into consideration. In fact, this is what we have done through Figure 8, 9, 10, 11, and 12. For details on how to implement the PFD and EFD policies see Section 3.

To fully understand these flowcharts, we defined the following parameters:

- *Arrival_num (AN)*: the output of Poisson random number generator which represents the total number of arrival packets during a slot time.
- *Arrival_counter (AC)*: counter for the total number of arrival packets for a frame.
- *Frame_counter*: counter for the number of arriving frames.
- *Sys_curr_state*: the current state of the system (i.e. the total number of packets in the system including the one that is being served).
- *Lost_fr_counter*: counter for the total number of corrupted frames.
- *Service_num (SN):* the output of Poisson random number generator which represents the total number of served packets during a slot time.
- *Sim_time*: simulation time that shows the number of slots to be simulated.
- *Buffer_size:* system capacity that indicates the maximum number of packets allowed to be stored in the system.
- *Lost_arr_counter:* the number of lost arriving packets for a frame (not accumulative) due to buffer overflow.
- *Flag:* flag for validity conditions and is set when *Lost_arr* is greater than or equal to one.
- *Lost_pcts_Accum*: counter for the total number of lost arriving packet accumulatively due to limited buffer size.
- *State_hist*: state histogram vector where *state_hist(n+1)* specifies the number of time slots the system stays in state *n* (where *n* packets are present (one packet at the server and *n*-1 packets in the buffer)).
- *Time_history*: time history vector where *time_history(i+1)* specifies system state at time slot *i*.
- *Total_arr_pcts*: total number of arrival packets that shows the total number of arriving packets to the system regardless if they are accepted or even discarded.
- *Num_of_packets*: the output of Geometric random number generator which represents the total number of packets for a certain frame.

It is interesting to realize that we can extract the PFD and EFD policies flowcharts from the NO CONTROL policy. Therefore, we would like to explain the major changes to these policies over the NO CONTROL policy.

For PFD policy: The following are the required changes over NO CONTROL policy to construct the PFD policy (we show these changes in Figure 15):

When the following condition that is shown in the NO CONTROL policy holds:

if *Arrival_counter < Num_of_packets and flag=1*

we do not need those blocks labeled at the right sides by numbers (1)-(4) in Figure 14. The reason for that is the flag is 1. Thus, there is a loss occurred to at least one of a certain frame packet. Therefore, the advantage of PFD policy will not permit us to store or accept those tailed packets to the buffer.

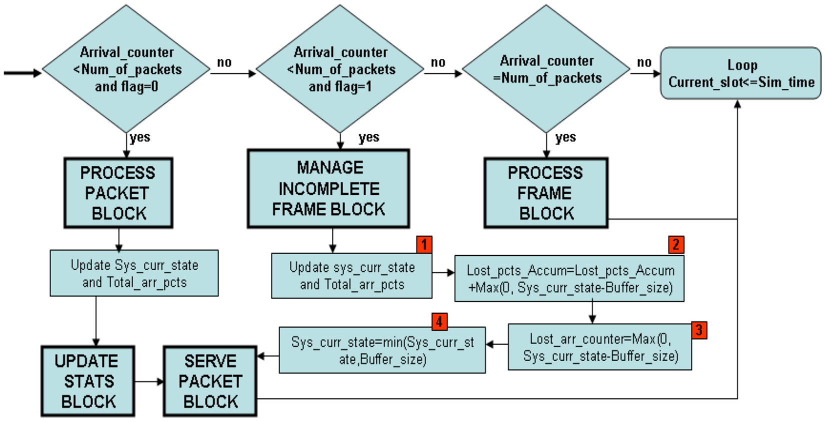

Figure 14 Major steps to implement the intermediate hop system when applying NO CONTROL (no QoD) policy

The first block (1) in Figure 14 is very important. The key is that it contains the update for *Sys_curr_state* by adding the current arriving packets (*Sys_curr_state* = *Sys_curr_state* + *Arrival_num*). Therefore, according to PFD policy, there is no update (addition) as long as these arriving packets belonging to the same useless frame. Since there is no acceptance under that condition, there is no need even for blocks (2)-(4) since the *Sys_curr_state* is not changed.

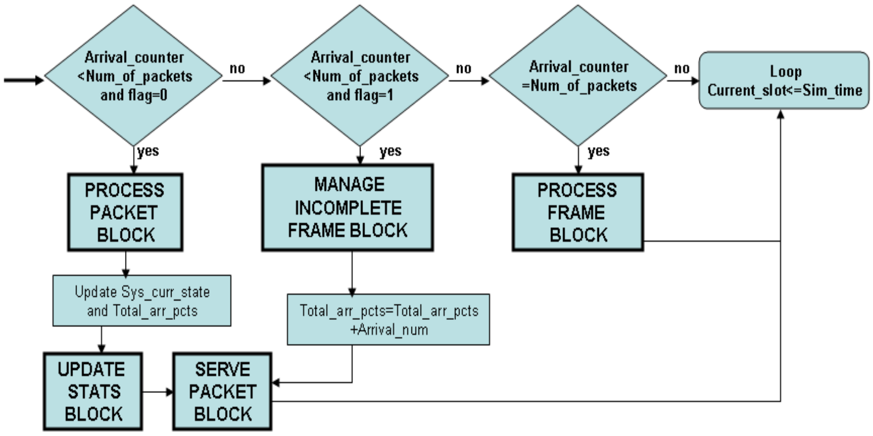

Figure 15 Major steps to implement the intermediate QoD-aware hop system when applying Partial Frame Discarding policy

For EFD policy: The following are the necessary changes over PFD policy to implement the EFD policy (we can see these changes in Figure 16):

The first change is after Process Frame Block. We define a variable called *reject_frame_flag*. In fact, this flag is important to reject a complete frame instead of tailed or rest of packets for a certain frame. Therefore, we define the following statements after Process Frame Block of course it does not matter before or after incrementing the *Frame_counter* by one:

if *Sys_curr_state>=K*

reject_frame_flag =1;

else *reject_frame_flag=0;*

The second change is regarding Update Statistics Block. This block is reachable under the following condition: *Arrival_counter < Num_of_packets and flag=0***.** Hence, the following condition should be implemented in block (6):

if *reject_frame_flag =1*

ignoreUpdate Statistics Block

else *update (consider them) as is*

Once more, ignoring is required due to no acceptance in the buffer (i.e. *Sys_curr_state* will remain as is). Thus, there is no need for Update Statistics Block, just go to the service immediately.

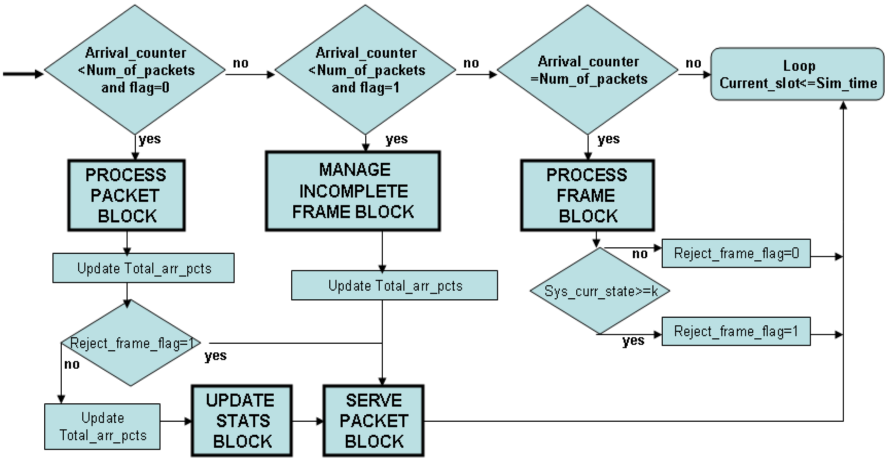

Figure 16 Major steps to implement the intermediate QoD-aware hop system when applying Early Frame Discarding policy

5.3 Simulation Results

We have run our simulations on a Pentium 4 3.0GHz with HT technology, 2MB L2 Cache, 800Mhz FSB, 512MB DDR Dual channel memory, 250 GB 7200RPM hard disk with 8MB cache system. The operating system is Windows XP. We have used MATLAB 7.0.0 for our simulations. The simulation time is 600000. The service rate is 1.5. We have generated 4 figures as an outcome. The simulations took 14 days when running two simulations in parallel.

In our simulations, we have measured the QoD in terms of frameput. Frameput is the ratio of frames that are served to the number of total frames:

$$frameput = \frac{\#\_of\_good\_frames}{\#\_of\_total\_frames}$$ where a good frame is a frame whose packets are received completely.

Figures 17, 18, and 19 show the frameput for NO CONTROL, PFD, and EFD (*K*=5) policies, respectively. Figure 20 shows the frameput result for EFD policy with a different threshold value and system capacity (*K*=10 and buffer size=20). We get the following 6 observations based on Figures 17, 18, 19, and 20.

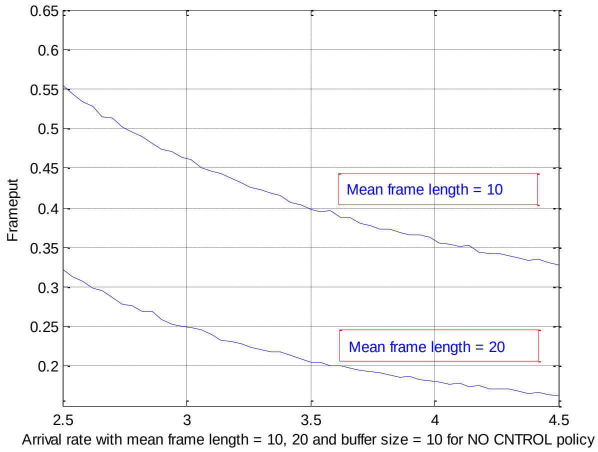

Figure 17 Frameput versus arrival rate for NO CONTROL policy, given the mean frame lengths are 10, 20 packets

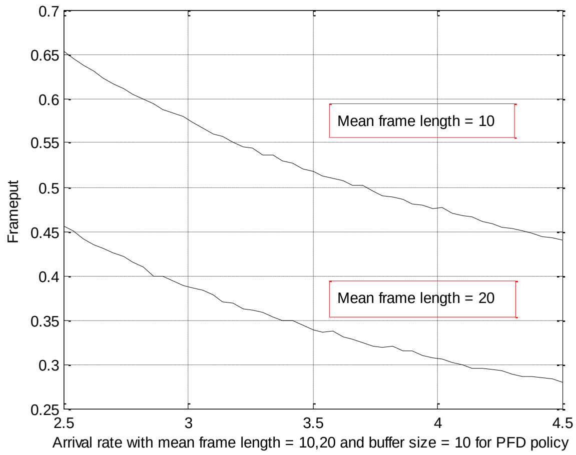

Figure 18 Frameput versus arrival rate for PFD policy, given the mean frame lengths are 10, 20 packets

1) *Frameput vs. Arrival rate.* The frameput decreases as the arrival rate increases for a fixed service rate, system capacity, and mean frame length. This is expected since an increase in arrival rate means an increase in the traffic (system flow) and getting lower probability to have good frames being served (i.e. blocking and frame loss probability is increased since there is fixed system capacity). Consequently, the accumulative number of good frames decreases with respect to the increase in the accumulative number of arriving frames. In fact, this supports our theoretical results, since decrease in frameput as arrival rate increases (for a fixed service rate and system capacity) means having high (increasing) average buffer size.

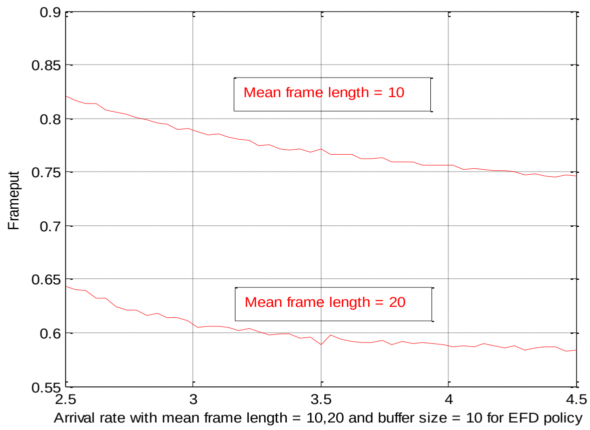

Figure 19 Frameput versus arrival rate for EFD policy, given the mean frame lengths are 10, 20 packets, and the threshold value (K=5)

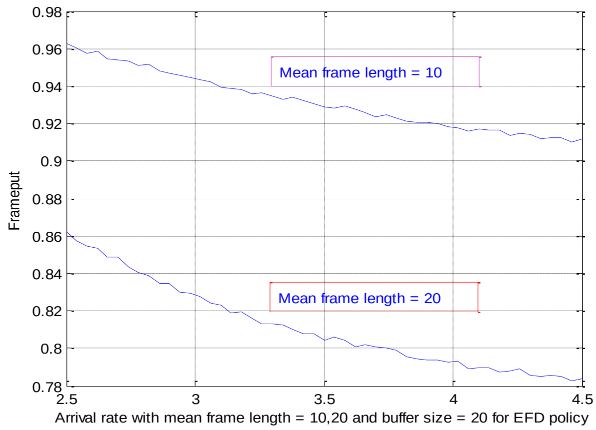

Figure 20 Frameput versus arrival rate for EFD policy, given the mean frame lengths are 10, 20 packets, and the threshold value (K=10)

2) *Frameput vs. Mean frame length.* The frameput decreases as the mean frame length increases for a fixed arrival rate, service rate, and system capacity. This is expected since increasing the mean frame size means getting low chances to serve good frames (i.e. the system throughput decreases) because of fewer and longer frames arriving at the router system. In addition, decreasing the mean frame length means there is a high probability for a head-of-frame packet (i.e. new frames) to arrive at the system. Thus, high chances to

serve more good frames. This can also validate our theoretical results for the average buffer occupancy since it increases as the mean frame length decreases.

3) *Frameput analysis for policies.* It is noticed from Figures 17, 18, and 19 that the frameput for PFD is better than frameput for NO CONTROL and the frameput for EFD is better than frameput for PFD. This shows that PFD policy is late in reacting to the congestion. Definitely these comparisons are for a fixed arrival rate, service rate, mean frame size, and system capacity. Actually this also validates our theoretical results for the average buffer occupancy since lower frameput means a high number of arriving corrupted frames. Thus, the average buffer occupancy for NO CONTROL is higher than in PFD policy, and it is higher in PFD policy than in EFD policy. When the arrival rate is 2.5 and mean frame size is 10, frameput for EFD (with K=5) is approximately 1.3 times better than the frameput of PFD policy. When the arrival rate is 4.5 and mean frame size is 10, frameput for EFD (with K=5) is approximately 1.7 times better than the frameput of PFD policy. For mean frame size 20, EFD (with K=5) performs 1.45 and 2.1 times better than the PFD policy at arrival rates 2.5 and 4.5, respectively. These results show that PFD policy cannot react to congestion for increasing arrival rates and mean frame sizes.

4) *Blocking probability analysis for policies.* The blocking probability for EFD policy is lower than the blocking probability in PFD policy. On the other hand, the blocking probability for PFD policy is lower than for blocking probability in NO CONTROL policy. Actually, as mentioned above, lower frameput means a lower number of good frames that indicates higher blocking probability. Therefore, worse QoD is obtained.

5) *Analysis of threshold and buffer size for EFD.* It can be seen from Figures 19 and 20 that the frameput for EFD policy increases as the system capacity increases for a fixed arrival rate, service rate, and mean frame length. This is expected since increasing the system capacity leads to having a greater number of good frames being served (i.e. decrease in the blocking probability). This supports our theoretical results in a sense that the average buffer occupancy increases as the system capacity increases. Thus, there is a good chance to have or serve more good frames.

6) *Validation with previous research.* Our simulation results validate the numerical [5] [6], analytical studies (or results) [7] [8], fluid studies [7] [19] [20] [21], and analytical study using discrete–time model [18] that have been done for the purpose of congestion avoidance through a packet level index (goodput). Thus, it is seen in our simulation results, that the frameput for EFD is better than frameput of PFD, and frameput for PFD is better than frameput in NO CONTROL. Actually, better means getting a higher number of good frames being served and a lower probability of dropping frame's packets. Consequently, lower congestion over the network is achieved.

6 CONCLUSION

The asymmetric rates for incoming and outgoing packet rates cause congestion if the outgoing (service) rate is more than the incoming. Since wireless hops (low bandwidth) are usually connected to a wired network (high bandwidth), the congestion for demanding applications like real-time video streaming is inevitable. In this paper, we have proposed a QoD-aware hop system that can reduce network congestion while maintaining acceptable QoD. We have provided our analysis mainly for three types of policies: No control, PFD, and EFD. Since we use a finite buffer size, the analysis of average buffer occupancy is important. We have first provided the theoretical analysis with closed form expressions for average buffer occupancy for both discarding policies. In our simulation, we have measure the QoD using frameput metric. Our results show that the EFD policy reacts to congestion better than PFD policy. No control policy is the worst among them. Consequently, the intermediate QoD-aware hop system is able to afford a double QoD improvement to customers by using an EFD policy with reasonable buffer sizes.

REFERENCES

[1] I. Cidon, A. Khamisy, and M. Sidi, "Analysis of packet loss processes in high speed networks," *IEEE Trans. Inform. Theory*, vol. 39, pp. 98–108, Jan. 1993.

[2] D. E. Comer, *"Internetworking with TCP/IP"*, vol. 1. Englewood Cliffs, NJ: Prentice-Hall, 1991.

[3] S. Floyd and V. Jacobson, "Random early detection gateways for congestion avoidance," *IEEE/ACM Trans. Networking*, vol. 1, no. 4, pp. 25–39, Aug. 1993.

[4] S. Floyd and A. Romanow, "Dynamics of TCP traffic over ATM networks," in *Proc. ACM/SIGCOMM'94*, pp. 79–88, Sept. 1994.

[5] Yael Lapid, Raphael Rom, and Moshe Sidi. "Analysis of Discarding Policies in High-Speed Networks". *IEEE Journal on Selected Areas in Communications*, 16 (5):764–777, June 1998.

[6] K. Kawahara, K. Kitajima, T. Takine, and Y. Oie, "Packet Loss Performance of Selective Cell Discard Schemes in ATM Switches". *IEEE Journal of Sel. Areas in Communications*, 15(5):903–913, June 1997.

[7] Dube Parijat and Altman Eitan. "Queueing and Fluid Analysis of Partial Message Discard Policy". In *proc. of 9th IFIP Working Conference on Performance Modeling and Evaluation of ATM and IP Networks*. Budapest, Hungary, June 2001.

[8] P. Dube and E. Altman, "Queueing analysis of early message discard policy**",** ICC 2002. IEEE International Conference on Communications, 2002.Volume 4, 28, 2426 – 2430, May 2002.

[9] D. Cross, and C.M. Harris, "Fundamentals of Queuing theory", John Wily and Sons. 1998.

[10] A.E. Kamal, "A performance study of selective cell discarding using the end-of-packet indicator in AAL type 5", In Proceeding of INFOCOM '95**,** pp. 1264-1272, Boston, April 1995.

[11] L. Kleinrock, "Queueing systems", vol. I: Theory. Wiley, New York, 1975.

[12] V. Markovski, F. Xue, and L.J. Trajkovic, ``Simulation and analysis of packet loss in video transfers using User Datagram Protocol," *The Journal of Supercomputing*, Kluwer, vol. 20, no. 2, pp. 175-196, Sept. 2001.

[13] M. Krunz, and H. Hughes, "A traffic for MPEG-coded VBR streams", Proc. of 1995 ACM SIGMETRICS Joint Int. Conf. on Measurement and Modeling of Computer Systems, Volume 23, No 1, 1995.

[14] H. M. Briceno, S. Gortler, and L. MacMillan, "NAIVE- Network aware Internet video encoding", Proc. of the 7th ACM Int. Conf. on Multimedia (Part 1), pp. 251-260, 1999

[15] D. Le Gall, "MPEG: A video compression standard for multimedia applications," Communications of the ACM, vol. 34, pp. 47-58, Apr. 1991.

[16] ITU-T Recommendation H.261, "Video Codec for audiovisual devices at p*64 kbits/s", Geneva 1990.

[17] A. Zhang, Y. Song, and M. Mielke, "NetMedia: streaming multimedia presentations in distributed environments", IEEE Multimedia, Volume 9, Issue 1, pp.56 – 73, March 2002.

[18] Wen-Hui Zhou; Ai-Hu Wang; Yong-Jun Li; "Analysis of a discrete-time queue for packet discarding policies in high-speed networks" Proceedings. 2005 IEEE Int. Conference on Wireless Communications, Networking and Mobile Computing, 2005. Volume 2, 23-26, Page(s):1083 – 1086.Sept. 2005.

[19] Parijat Dube, Eitan Altman, "On Fluid Analysis of Queues with Selective Message Discarding Policies," in the proc. of 38th Annual Allerton Conference on Communication, Control and Computing, Oct 4-6, 2000, Monticelio, Illinois, USA.

[20] Parijat Dube, Eitan Altman, "Fluid Analysis of Early Message Discard Policy Under Heavy Traffic," proc. of INFOCOM 2002, June 23-27, 2002, New York City, USA.

[21] Parijat Dube, Eitan Altman, "Goodput Analysis of a Fluid Queue with Selective Burst Discarding and a Responsive Bursty Source", in the proc. of INFOCOM 2003, April 1-3, 2003, San Francisco, CA, USA.

[22] S. Yi, M. Kappes, S. Garg, X. Deng, G. Kesidis, and C. Das, "Proxy-RED: an AQM scheme for wireless local area networks." *Wirel. Commun. Mob. Comput.* 8, 4 (May. 2008), 421-434.

[23] James E. Gentle, "Random Number Generation and Monte Carlo Methods", Statistics and Computing Series, Springer, 2002

[24] S. Yi, M. Kappes, S. Garg, X. Deng, G. Kesidis, and C. Das, "Proxy-RED: an AQM scheme for wireless local area networks", Wireless Communications and Mobile Computing (in press)

[25] K. A. Darabkh, and R. S. Aygun, "Simulation of Quality of Service of Congestion Control in Multimedia Networking Using Frame Discarding Policies" *Proceedings of* 2006 SCS International Conference on Modeling and Simulation - Methodology, Tools, Software Applications (M&S-MTSA'06), Calgary, Canada

[26] Khalid Darabkh and Ramazan Savas Aygun. "Quality of Service Evaluation of Congestion Control for Multimedia Networking" 2006 International Conference on Internet Computing, Las Vegas, June 26-29, 2006.

[27] Tim Farnham, Mahesh Sooriyabandara, and Costas Efthymiou, "Enhancing multimedia streaming over existing wireless LAN technology using the Unified Link Layer API," International Journal of Network Management, John Wiley & Sons Ltd, Volume 17, Issue 5, pp. 331–346, Sept. 2007.

[28] Julian Chesterfield, Rajiv Chakravorty, Ian Pratt, Suman Banerjee, and Pablo Rodriguez, "Exploiting Diversity to Enhance Multimedia Streaming Over Cellular Links," In Proceedings of IEEE INFOCOMM '05, pp. 2020 – 2031, Miami, USA, 2005.

[29] Shuang Li, Raghu Kisore Neelisetti, Cong Liu, Santosh Kulkarni, and Alvin Lim, "An Interference-Aware Routing Algorithm for Multimedia Streaming Over Wireless Sensor Networks," The International journal of Multimedia & Its Applications (IJMA), Volume 2, Number 1, pp.10-30, Feb. 2010.

[30] W. Zhou and Y. Li, "A discrete-time queuing model of EMD policy in high-speed networks", Applied Mathematics and Computation, Elsevier Inc, Volume 181, Issue 1, pp. 543–551, Oct. 2006.

[31] K.A. Darabkh and R. S. Aygün, "TCP Traffic Control Evaluation and Reduction over Wireless Networks Using Parallel Sequential Decoding Mechanism," *EURASIP Journal on Wireless Communications and Networking*, vol. 2007, Article ID 52492, 16 pages, 2007.

[32] Ye Tian, Kia Xu, and N. Ansari, "TCP in wireless environments: Problems and Solutions," *Communications Magazine*, IEEE, Volume 43, Issue 3, pp. S27 - S32, Mar. 2005.

[33] H. Balakrishnan, V. N. Padmanabhan, and R. H. Katz, “The Effects of Asymmetry on TCP Performance,” *Proc. ACM/IEEE Mobicom*, pp. 77–89, Sept. 1997.

[34] A. Trad, Q. Ni, and H. Afifi, “Adaptive VoIP Transmission over Heterogeneous Wired/Wireless Networks,” Interactive Multimedia and Next Generation Networks, Lecture Notes in Computer Science, pp. 25-36, 2004.

[35] A Elyousfi, A Tamtaoui, and E Bouyakhf, “A new fast intra prediction mode decision algorithm for **H. 264**/AVC encoders,” International Journal of Electrical, Computer, and Systems Engineering 4:1 2010