

# End-to-end QoS for Differentiated Services and ATM Internetworking<sup>1</sup>

Hongjun Su    Mohammed Atiquzzaman  
Dept. of Electrical Computer Engineering  
University of Dayton, Dayton, OH 45469-0226  
Email: suhongju@flyernet.udayton.edu    atiq@udayton.edu

*Abstract*—The Internet was initially design for non real-time data communications and hence does not provide any Quality of Service (QoS). The next generation Internet will be characterized by high speed and QoS guarantee. The aim of this paper is to develop a prioritized early packet discard (PEPD) scheme for ATM switches to provide service differentiation and QoS guarantee to end applications running over next generation Internet. The proposed PEPD scheme differs from previous schemes by taking into account the priority of packets generated from different application. We develop a Markov chain model for the proposed scheme and verify the model with simulation. Numerical results show that the results from the model and computer simulation are in close agreement. Our PEPD scheme provides service differentiation to the end-to-end applications.

*Keywords*—Differentiated Services, TCP/IP-ATM Internetworking, End-to-end QoS, Queue analysis, analytical model, performance evaluation, Markov chains.

## I. INTRODUCTION

With quick emergence of new Internet applications, efforts are underway to provide Quality of Service (QoS) to the Internet. Differentiated Services (DS) is one of the approaches being actively pursued by the Internet Engineering Task Force (IETF) [1], [2], [3], [4], [5]. It is based on service differentiation, and provides aggregate services to the various application classes. DS has defined three service classes. When running DS over ATM (which is implemented by many Internet service providers as their backbones), we need proper services mapping between them. Premium Service requires delay and loss guarantees, and hence it can be mapped to the ATM Constant Bit Rate (CBR) service. Assured Service only requires loss guarantees and hence can be mapped to ATM Unspecified Bit Rate (UBR) service with Cell Loss Priority (CLP) bit set to zero. The Best Effort service does not require any loss or delay guarantee and can be mapped to the ATM UBR service with CLP bit set to one.

It has been shown that Internet may lose packets during high load periods, even worse is that it may suffer congestion collapse [6], [7]. Packets loss means all of the resources they have consumed in transit are wasted. When running DS over ATM, packets loss may lead to more serious results. Because messages will be break into small fix size packet (call cells), one packet loss will lead to the whole message be transmitted again [8]. This makes the congestion scenario even worse. Transmitting useless incomplete packets in a congested network wastes a lot of resource and may result in a very low goodput (good throughput) and poor bandwidth

utilization of the network. A number of message based discard strategies have been proposed to solve this problem [8], [9], [10], [11]. These strategies attempt to ensure that the available network capacity is effectively utilized by preserving the integrity of transport level packets during congestion periods. Early Packet Discard (EPD) strategy [8] drops entire messages that are unlikely to be successfully transmitted prior to buffer overflow. It prevents the congested link from transmitting useless packets and reduces the total number of incomplete messages. EPD achieves this by using a threshold in the buffer. Once the queue occupancy in the buffer exceeds this threshold, the network element will only accept packets that belong to a message that has at least one packet in the queue or has already been transmitted. Also per-VC based EPD schemes [12], [13] are proposed to solve the fairness problem that a pure EPD may suffer when virtual circuits compete for the resource. Although EPD can improve the goodput at a network switch, it does not distinguish among priorities of different applications. Previous studies on EPD have assumed a single priority of all ATM packets, and thus fail to account for the fact that ATM packets could have priority and need to be treated differently. Without a differentiation between the packets, end-to-end QoS guarantee and service differentiation promised by DS networks cannot be ensured when packets traverse through an ATM network. The objective of this study is to developed message based discarding scheme which will account for priority of packets and will be able to provide service differentiation to end applications.

In this paper, we propose a prioritized EPD (PEPD) scheme which can provide the necessary service differentiation needed by the future QoS network. In the PEPD scheme, two thresholds are used to provide service differentiation. We have developed Markov chain models to study the performance of our proposed scheme. The effectiveness of PEPD in providing service differentiation to the two classes of ATM packets coming from a DS network is estimated by the model and then validated by results obtained from our simulation. We measure the goodput, packet loss probability and throughput of the two service classes as a function of the load. Given a QoS requirement for the two service classes, our model can predict the size of the buffer required at the ATM switches and the value of the two thresholds to be used to achieve the target QoS. This model can provide a general framework for analysis of networks carrying messages from applications which require differen-

<sup>1</sup>This work was supported by NASA grant no. NAG3-2318 and Ohio Board of Regents Research Challenge grant

tial treatment in terms of Quality of Service (QoS).

The rest of this paper is organized as follows. Section II lists the assumptions used in the model. Section III constructs a Markov chain model to analyze our proposed PEPD scheme. The model is used to study the performance of the PEPD policy using goodput as the performance criteria. Numerical results from both modeling and computer simulation are presented in Section IV. Concluding remarks are given in Section V.

## II. MODELING ASSUMPTIONS

In the dispersed message model [11], [14], a higher layer protocol data unit (message) consists of a block of consecutive packets that arrive at a network element at different time instants. TCP/IP based systems are examples of such a model. In TCP/IP, the application message is segmented into packets, which are then transmitted over the network. At the receiving end, they are reassembled back into a message by the transport protocol before being delivered to higher layers.

- We assume variable length packets, the length of the packets being geometrically distributed with parameter  $q$  (independent between subsequent packets). Clearly, the average packet length is  $1/q$  packets. This kind of assumption is typical for data application such as document file and e-mail.
- We also assume that the packets arrive at a network element according to a Poisson process with rate  $\lambda$ , and the transmission time of a packet is exponentially distributed with rate  $\mu$ . Although we assume that packets are of variable length, Lapid's work [11] shows that this kind of model fits well for fixed-length packet (which is typical to ATM network) scenarios.
- The network element we used in this paper is a simple finite input queue that can contain at most  $N$  packets. When the packets arrive at the network element, it enters the input queue only when there is space to hold it; otherwise it is discarded.
- Packets leave the queue according to a first-in-first-out (FIFO) order. When a server is available, the packet at the head of the queue can be served. A packet is transmitted by the server of the network element during its service time. Hence, the network element can be viewed as a  $M/M/1/N$  model, with arrival rate  $\lambda$  and service rate  $\mu$ .
- The input load to the network element is defined as  $\rho = \lambda/\mu$ .

## III. PERFORMANCE EVALUATION OF PEPD SCHEME

In this section, we describe the PEPD scheme, followed by model setup and performance analysis.

### A. PEPD scheme

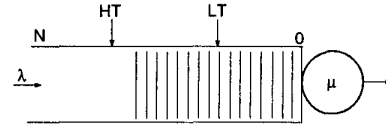


Fig. 1. Network element using PEPD policy.

In the PEPD scheme, we use two thresholds: a low threshold ( $LT$ ) and a high threshold ( $HT$ ), with  $0 \leq LT \leq N$ ,  $LT \leq HT \leq N$ . As shown in Fig. 1, let  $QL$  indicate the current queue length. The following strategy is used to accept packets in the buffer.

- If  $QL < LT$ , all packets are accepted in the buffer.
- If  $LT \leq QL < HT$ , new low priority messages will be discarded; only packets belonging to new messages with high priority or packets belonging to messages which have already entered the buffer are accepted.
- If  $HT \leq QL < N$ , all new messages of both priorities are discarded.
- For  $QL \geq N$ , packets belonging to all messages are lost because of buffer overflow.

### B. Proposed PEPD model

To model the PEPD scheme, we must distinguish between two modes: the *normal mode* in which packets are accepted and the *discarding mode* in which arriving packets are discarded. The state transition diagram for this policy is shown in Fig. 2. In the diagram, state  $(i, j)$  indicates that the buffer has  $i$  packets and is in  $j$  mode, where  $0 \leq i \leq N$ ,  $j = 0$  or  $1$ .  $j = 0$  corresponds to the normal mode, while  $j = 1$  represents the discarding mode. We assume that a head-of-message packet arrives with probability  $q$ . The probability that an arriving packet is part of the same message as the previous packets is  $p = 1 - q$ , and hence is discarded with that probability in the case that that message is being discarded.

According to PEPD, if a message starts to arrive when the buffer contains more than  $LT$  packets, the complete new message is discarded if it is of low priority, while if a new message starts to arrive when the buffer contains more than  $HT$  packets, the complete message is discarded regardless of its priority. Once a packet is discarded, the buffer enters the discarding mode, and discards all packets belonging to this discarded message. The system will remain in discarding mode until another head-of-message packet arrives. If this head-of-message packet arrives when  $QL < LT$ , it is accepted, and the system enters the normal mode. If this packet arrives when  $LT \leq QL \leq HT$ , then the system enters the normal mode only if this packet has high priority. Otherwise, it stays in the discarding mode. Of course, when  $QL > HT$ , the buffer stays in the discarding mode. Let's assume that  $h$  and  $l = 1 - h$  be the probabilities of a message being of high and low priority respectively. Also let

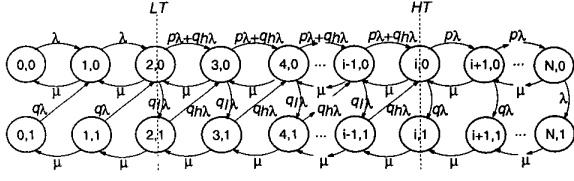


Fig. 2. Steady-state transition diagram the buffers using PEPD

$P_{i,j}$  ( $0 \leq i \leq N, j = 0, 1$ ) be the steady-state probability of the buffer being in state  $(i, j)$ . From Fig. 2, we can get the following equations. The solutions of these equations will generate the steady-state probabilities of the buffer states.

$$\begin{aligned}
\lambda P_{0,0} &= \mu P_{1,0} \\
q\lambda P_{0,1} &= \mu P_{1,1} \\
(\lambda + \mu)P_{i,0} &= \lambda P_{i-1,0} + \mu P_{i+1,0} + qh\lambda P_{i-1,1} \\
&\quad 1 \leq i \leq LT \\
(\lambda + \mu)P_{i,0} &= (\lambda p + qh\lambda)P_{i-1,0} + \mu P_{i+1,0} \\
&\quad + qh\lambda P_{i-1,1} \quad LT < i \leq HT \\
(\lambda + \mu)P_{i,0} &= p\lambda P_{i-1,0} + \mu P_{i+1,0} \quad HT < i < N \\
(\lambda + \mu)P_{N,0} &= p\lambda P_{N-1,0} \\
\mu P_{N,1} &= \lambda P_{N,0} \\
\mu P_{i,1} &= q\lambda P_{i,0} + \mu P_{i+1,1} \quad HT \leq i < N \\
(qh\lambda + \mu)P_{i,1} &= q\lambda(1-h)P_{i,0} + \mu P_{i+1,1} \\
&\quad LT \leq i < HT
\end{aligned} \tag{1}$$

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

### C. Performance analysis of PEPD

In this section, we derive the expression of goodput  $G$  for high and low priority messages. The goodput  $G$  is the ratio between total good packets exiting the buffer and the total arriving packets at its input. Good packets are those packets that belong to a complete message leaving the buffer. In this paper, we define the goodput for high (or low) priority as the ratio between total number of good packets with high (or low) priority exiting the system and the total number of arriving high (or low) priority packets at the buffer. However, we normalize the goodput to the maximum possible goodput.

Let  $W$  be the random variable that represents the length (number of packets) of an arriving message, and  $V$  be the random variable that represents the success of a message.  $V = 1$  for a good message, and  $V = 0$  for an incomplete message. Let  $U$  be the random variable that represents the priority of a packet,  $U = 1$  for high priority packets and  $U = 0$  for low priority packets. The goodput for the high

priority packets ( $G_h$ ) is

$$G_h = \frac{\sum_{n=1}^{\infty} nP(W = n, V = 1, U = 1)}{\sum_{n=1}^{\infty} nP(W = n, U = 1)} \tag{2}$$

where the numerator represents the total good packets exiting the buffer and the denominator is the total arriving packets at a network input. Note that  $W$  and  $V$  are independent random variables, and the length of an arriving message is geometrically distributed with parameter  $q$ , which means the average length of the messages is  $1/q$ . Then the denominator of Eq. (2) can be expressed as

$$\sum_{n=1}^{\infty} nP(W = n, V = 1) = P(U = 1) \sum_{n=1}^{\infty} P(W = n) = \frac{h}{q} \tag{3}$$

Substituting Eq. (3) in Eq. (2),

$$G_h = \frac{q}{h} \sum_{n=1}^{\infty} nP(W = n, V = 1, U = 1) \tag{4}$$

The probability of an incoming high priority message of length  $n$  to be transmitted successfully can be expressed as follows:

$$\begin{aligned}
P(W = n, V = 1, U = 1) &= P(V = 1|W = n, U = 1) \\
&\quad P(W = n, U = 1) \\
&= P(V = 1|W = n, U = 1) \\
&\quad P(W = n)P(U = 1) \tag{5} \\
&= q(1-q)^{n-1}h \\
&\quad P(V = 1|W = n, U = 1)
\end{aligned}$$

Let  $Q$  be the random variable representing the queue occupancy at the arrival of a head-of-message packet. Then

$$\begin{aligned}
P(V = 1|W = n, U = 1) &= \sum_{i=0}^N P(V = 1|W = n, U = 1, \\
&\quad Q = i)P(Q = i) \tag{6}
\end{aligned}$$

where  $P(Q = i) = P_{i,0} + P_{i,1}$  is the probability of the queue occupancy.  $P_{i,j}$  is obtained from the solution of Eq. (2). By combining Eqs. (4), (5), and (6), we get the goodput of the high priority messages as:

$$\begin{aligned}
G_h &= q \sum_{n=1}^{\infty} nq(1-q)^{(n-1)} \sum_{i=0}^N P(V = 1|W = n, U = 1, \\
&\quad Q = i)P(Q = i) \tag{7}
\end{aligned}$$

Similarly, we can get the goodput for the low priority messages and the total goodput as follows:

$$\begin{aligned}
G_l &= q \sum_{n=1}^{\infty} nq(1-q)^{(n-1)} \sum_{i=0}^N P(V = 1|W = n, U = 0, \\
&\quad Q = i)P(Q = i) \tag{8} \\
G &= q \sum_{n=1}^{\infty} nq(1-q)^{(n-1)} \sum_{i=0}^N P(V = 1|W = n, \\
&\quad Q = i)P(Q = i) \tag{9}
\end{aligned}$$

In order to find the values of  $G$ ,  $G_l$  and  $G_h$ , we need to define and evaluate the following conditional probabilities:

$$S_{n,i} = P(V = 1|W = n, Q = i) \quad (10)$$

$$S_{l,n,i} = P(V = 1|W = n, U = 0, Q = i) \quad (11)$$

$$S_{h,n,i} = P(V = 1|W = n, U = 1, Q = i) \quad (12)$$

These conditional probabilities can be computed recursively. Let's take  $S_{h,n,i}$  as an example. Consider first a system that employs the PPD policy. Usually, the success of a packet depends on the evolution of the system after the arrival of the head-of-message. However, there is a boundary condition for this. Let us first consider a message of length  $1 \leq n \leq N$ . Assume that the head-of-message packet belonging to a message of length  $n \leq N$  arrives at buffer when  $Q = i$ . Then, if  $i \leq N - n$ , there is enough space to hold this message, and this message is guaranteed to be good, i.e.

$$\hat{S}_{n,i} = 1 \quad 0 \leq i \leq N - n, \quad 1 \leq n \leq N \quad (13)$$

note that if  $Q = N$  (i.e. the buffer is full), the head-of-message packet is discarded, and the message is guaranteed to be bad. Hence,

$$\hat{S}_{n,N} = 0 \quad 1 \leq n \leq N \quad (14)$$

Eqs. (13) & (14) give the boundary conditions for this system. For other states of the buffer, we have:

$$\hat{S}_{n,i} = (1 - r)\hat{S}_{n-1,i+1} + r\hat{S}_{n,i-1} \quad N - n + 1 \leq i \leq N - 1, 1 \leq n \leq N \quad (15)$$

where  $r = \mu/(\mu + h\lambda)$  is the probability that a departure occurs before an arrival. In this case, we only consider the conditional probability for high priority packets, so the arrival rate is  $h\lambda$  rather than  $\lambda$ . Eq. (15) can be explained as follows. If the next event following the arrival of a head-of-message packet is the arrival of another packet (which has the probability  $1 - r$ ), this new packet can be viewed as a new head-of-message packet belonging to a message of length  $n - 1$ . Therefore, the probability that this new message will succeed is  $\hat{S}_{n-1,i+1}$ . If the event following the arrival of the head-of-message packet is a departure of a packet (which happens with probability  $r$ ), the probability that the message is successful is  $\hat{S}_{n,i-1}$ , since it is equivalent to a head-of-message packet that arrived at the system with  $Q = i - 1$  packets. So, combining the above two conditions, we can get:

$$\hat{S}_{n,i} = \begin{cases} 1 & N - n + 1 \leq i \\ (1 - r)\hat{S}_{n-1,i+1} + r\hat{S}_{n,i-1} & i \leq N - 1 \\ 0 & i = N \end{cases} \quad (16)$$

For a large message,  $n > N$ , there is no guarantee that this message will succeed, it's success depended heavily on

the evolution of the system after the arrival of the head-of-message packet even for the case of  $i = 0$ . So, for  $n > N$  we get the following equations:

$$\hat{S}_{n,i} = \begin{cases} (1 - r)\hat{S}_{n-1,i+1} + r\hat{S}_{n-1,i} & i = 0 \\ (1 - r)\hat{S}_{n-1,i+1} + r\hat{S}_{n,i-1} & N - n + 1 \leq i \\ 0 & i \leq N - 1 \\ 0 & i = N \end{cases} \quad (17)$$

These recursions are computed in ascending order of both  $n$  and  $i$ . For a system that employs the PEPD policy, for high priority messages, the above recursions remain correct only when the head-of-message packet arrives at the buffer while the number of packets is below the high threshold, i.e.  $Q = i < HT$ . For  $Q = i \geq HT$ , these new messages will be discarded, so

$$S_{h,n,i} = \begin{cases} \hat{S}_{n,i} & i < HT \\ 0 & HT \leq i \leq N \end{cases} \quad (18)$$

with  $r = \mu/(\mu + h\lambda)$ . Similarly, we can get

$$S_{l,n,i} = \begin{cases} \hat{S}_{n,i} & i < LT \\ 0 & LT \leq i \leq N \end{cases} \quad (19)$$

with  $r = \mu/(\mu + (1 - h)\lambda)$ , while the average is

$$S_{n,i} = (1 - h)\hat{S}_{l,n,i} + h\hat{S}_{h,n,i} \quad (20)$$

The above model is used to analyze the performance of PEPD in the next section.

#### IV. NUMERICAL RESULTS

In this section, we present results from our analytical model and simulation to illustrate the performance of PEPD. We also validate the accuracy of our analytical model by comparison with simulation results. In our experiment, we set  $N = 120$  packets,  $q = 1/6$  which corresponds to the case where the queue size is 20 times the mean message length. The incoming traffic load ( $\rho$ ) at the input to the buffer is set in the range of  $0.8 - 2.2$ , where  $\rho < 1$  represents moderate load, and  $\rho \geq 1$  corresponds to higher load which results in congestion buildup at the buffer. Goodput of the combined low and high priority packets is defined as  $G = h * GH + (1 - h) * GL$  as used in Eq. (20).

In order to validate our model, we compare it with results from computer simulation. The simulation setup is simply two nodes compete for a single link with a queue size 120 packets. The two nodes generate messages with a mean length of 6 (measured in packets). Because the queue occupancy is a critical parameter used for calculating the goodput, we compare the queue occupancy obtained from the model and computer simulation in Fig. 3. For  $q = 1/6$ , it is clear that analytical and simulation results are in close

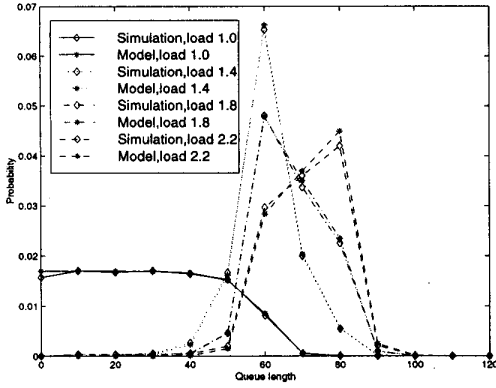


Fig. 3. Comparison of queue occupancy probability from model and simulation with different load.  $N = 120, LT = 60, HT = 80, h = 0.5, q = 1/6$ .

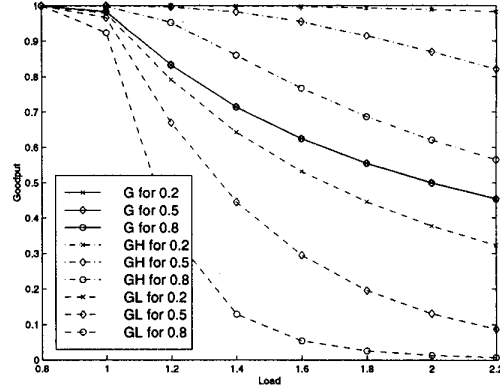


Fig. 5. Goodput versus load for  $h = 0.2, 0.5$  and  $0.8$  with  $N = 120, LT = 60, HT = 80, q = 1/6$ . G, GH and GL represents average goodput of all, high priority, and low priority packets respectively.

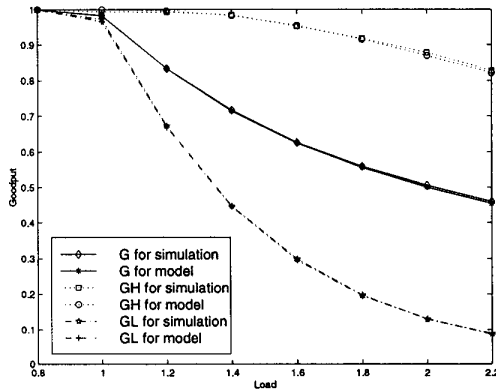


Fig. 4. Goodput versus load for  $h = 0.5, N = 120, LT = 60, HT = 80, q = 1/6$ . G, GH and GL represents average goodput of all, high priority, and low priority packets respectively.

agreement. Our proposed scheme results in the buffer occupancy varying between  $LT$  and  $HT$  for even high loads. The exact value depends on the average message length, queue thresholds, etc.

Fig. 4 shows the goodput of the buffer using PEPD for  $q = 1/6$  (i.e. mean message length of 6) as a function of the offered load. In this figure, the probability that a message is of high priority is 0.5. From Fig. 4, it is clear that the results from our model and computer simulation fit well. So we conclude that our model can be used to carry out an accurate analysis the PEPD policy. Therefore, in the rest of this section, we will use results from only the model to analyze the performance of PEPD policy.

Fig. 5 shows the goodput for  $q = 1/6$  as a function of the offered load and for different mix ( $h$ ) of high & low priority packets. For a particular load, increasing the fraction of High Priority (HP) packets ( $h$ ) results in a decrease of throughput of both high and Low Priority (LP) packets. The LP throughput decreases because the increase in  $h$  results

in fewer LP packets at the input to the buffer in addition to LP packets competing with more HP packets in the buffer space (0 to  $LT$ ). On the other hand, increase in  $h$  results in more HP packets. Since the amount of buffer space ( $HT-LT$ ) which is reserved for HP packets is the same, the throughput of HP packets decrease. Note that the decrease in the throughput of LP is much faster than the decrease in HP resulting in the overall goodput (as defined by Eq. (20)) being constant. Our proposed technique allows higher goodput for high priority packets which may required in scenarios where an application may need a preferential treatment over other applications.

In Fig. 6, we fix  $LT$  while varying  $HT$  to observe the behavior of the buffer. It is obvious that for a traffic containing fewer high priority packets, increasing the  $HT$  will increase the performance of the buffer for high priority packets. This is because increasing  $HT$  will let the high priority packets get more benefits from discarding low priority packets, especially for lower values of  $HT$ . Increasing  $HT$  will result in an initial increase in the goodput for high priority packets followed by a decrease. This is obvious, because for a very high value of  $HT$ , the behavior of PEPD will approach that of PPD for high priority packets.

Fig. 7 shows the goodput for high priority message versus the fraction of high priority messages. It is also clear that for a particular load, increasing the high priority traffic will decrease the performance for high priority packets as has been observed in Fig 5.

Finally, in Fig. 8, we keep  $HT$  constant while changing  $LT$ . For a load of 1.6 and a particular mix of high & low priority packets, we observe that the performance of high priority packets is not very sensitive to a change in  $LT$ . However, when  $LT$  is set close to  $HT$ , the goodput for high priority packets will decrease quickly. This is because when the two thresholds are set too close, the high priority packets do not get enough benefits from discarding low priority packets. We suggest avoiding this mode of operation because the

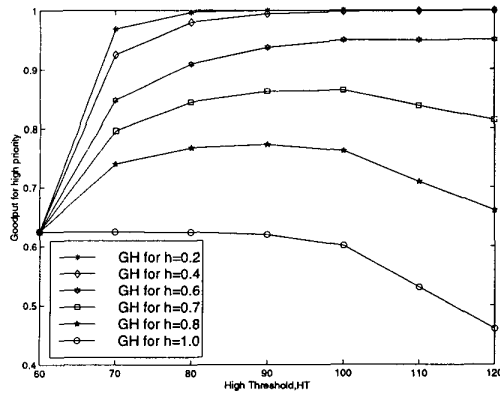


Fig. 6. Goodput for high priority messages versus  $HT$  for different  $h$  with  $LT = 60$ ,  $q = 1/6$  and input load of 1.6

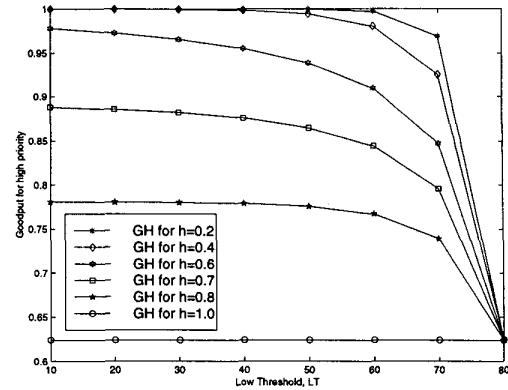


Fig. 8. Goodput for high priority versus  $LT$  for  $N = 120$ ,  $HT = 80$ ,  $load=1.6$ ,  $q = 1/6$ .

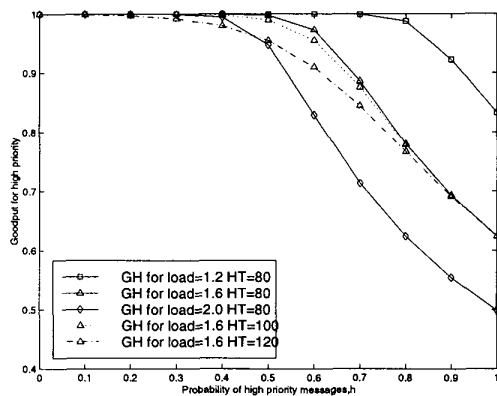


Fig. 7. Goodput of high priority messages versus  $h$  for different load and  $HT$  for  $LT = 60$ .

buffer is not fully utilized.

## V. CONCLUSIONS

In this paper, we have proposed and developed a performance model for the Priority based Early Packet Discard (PEPD) to allow end to end QoS differentiation for applications over Next Generation Internet. To verify the validity of our proposed analytical model, we compared it with results from computer simulation. Numerical results show that the results from the model and computer simulation are in close agreement. The numerical results also show that our proposed PEPD policy can provide differential QoS to low and high priority packets. Such service differentiation is essential to provide QoS to applications running Differentiated service over ATM. Our result show that the performance of PEPD depends on the mix of high & low priority traffic, threshold setting, average message length, etc. Given a certain QoS, the model can be used to dimension the size of the buffer and the PEPD thresholds. Our model can

serve as a framework to implement packet based discard schemes using priority. Results show that this scheme solves some critical problems for running Differentiated Service (DS) over ATM network by ensuring the QoS promised by the Differentiated Service.

## REFERENCES

- [1] S. Blake, D. Black, M. Carlson, Z. Wang, and W. Weiss, "An architecture for differentiated services." RFC 2474, Internet Engineering Task Force, December 1998.
- [2] K. Nichols, V. Jacobson, and L. Zhang, "A two-bit differentiated services architecture for the internet." RFC 2638, Internet Engineering Task Force, July 1999.
- [3] K. Nichols, S. Blake, F. Baker, and D. Black, "Definition of the differentiated services field (DS Field) in the IPv4 and IPv6 headers." RFC 2474, Internet Engineering Task Force, December 1998.
- [4] J. Heinanen, F. Blaker, W. Weiss, and J. Wroclawski, "Assured forwarding PHB group." RFC 2597, Internet Engineering Task Force, June 1999.
- [5] X. Xiao and L. M. Ni, "Internet QoS: A big picture," *IEEE Network*, pp. 8–18, April 1999.
- [6] J. Nagle, "Congestion control in IP/TCP internetworks." RFC 896, Internet Engineering Task Force, January 1984.
- [7] V. Jacobson, "Congestion avoidance and control," *Proceedings of SIGCOMM'88*, pp. 314–329, August 1988.
- [8] A. Romanow and S. Floyd, "Dynamics of TCP traffic over ATM network," *IEEE Journal on Selected Areas in Communications*, vol. 13, pp. 633–641, 1995.
- [9] K. Kawahara, K. Kitajima, T. Takine, and Y. Oie, "Packet loss performance of the selective cell discard schemes in ATM switches," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 903–913, 1997.
- [10] M. Casoni and J. S. Turner, "On the performance of early packet discard," *IEEE Journal on Selected Areas in Communications*, vol. 15, pp. 892–902, 1997.
- [11] Y. Lapid, R. Rom, and M. Sidi, "Analysis of discarding policies in high-speed networks," *IEEE Journal on Selected Areas in Communications*, vol. 16, pp. 764–772, 1998.
- [12] H. Li, K. Y. Siu, H. Y. Tzeng, H. Ikeda, and H. Suzuki, "On TCP performance in ATM networks with per-VC early packet discard mechanisms," *Computer Communications*, vol. 19, pp. 1065–1076, 1996.
- [13] J. S. Turner, "Maintaining high throughput during overload in ATM switches," *IEEE INFOCOM '96: Conference on Computer Communications*, San Francisco, USA, pp. 287–295, Mar. 26–28 1996.
- [14] I. Cidon, A. Khamisy, and M. Sidi, "Analysis of packet loss processes in high speed networks," *IEEE Transactions on Information Theory*, vol. 39, pp. 98–108, 1993.