

# A Performance Model of Partial Packet Discard and Early Packet Discard Schemes in ATM Switches\*

Hairong Sun, Xinyu Zang and Kishor S. Trivedi  
{hairong, xzang, kst@ee.duke.edu}  
Center for Advanced Computing and Communications  
Department of Electrical and Computer Engineering  
Duke University  
Durham, NC 27708

## Abstract

In this paper, we develop a concise performance model of Partial Packet Discard (PPD) and Early Packet Discard (EPD) scheme in ATM switches. We study the performance of PPD and EPD with heterogeneous traffic sources. The sources included Poisson, and ON-OFF with long-tailed sojourn time distribution which is approximated by a hyperexponential distribution. The fairness of EPD is investigated. We automatically generate and numerically solve the underlying Markov chain using a high-level graphical paradigm known as the stochastic reward net. Our numerical results reveal that: (1) The benefit of PPD and EPD is not significant when the queueing system is underloaded with Poisson sources. (2) PPD and EPD can increase the goodput when the system is overloaded or loaded with ON-OFF sources. (3) In *All Poisson* case, PPD and EPD provide nearly fair service to the sources. (4) The ON-OFF source gets higher goodput than the Poisson source. Because the burstiness of a source will be alleviated by the statistical multiplexing of ATM switches, the ON-OFF source may be viewed as the source that just enters the network, and the Poisson source may be viewed as the source being far away from the network node we are considering and has been regulated and smoothed by the switches it traversed. Therefore the queueing system with EPD algorithm gives higher goodput to the sources near it. And the sources far away from the system will have lower goodput. One of our principal conclusion is that per VC-based scheme is not needed at the core of large ATM networks because the traffic

---

\*This research was supported in part by the National Science Foundation under Grant No. EEC9418765, and by the North Carolina Advanced Network Program as an enhancement Project to the Center for Advanced Computing and Communications in Duke University.

is less bursty at the core of the networks and PPD and EPD are fair under this kind of environment. A per VC-based scheme may be used at the edge of the ATM cloud. Such a configuration can make the core of the ATM network work at high speed.

**Index Terms:** ATM, Early Packet Discard, Stochastic Reward Net, Stochastic Petri Net Package (SPNP)

# 1 Introduction

While ATM (Asynchronous Transfer Mode) was originally conceived as a carrier of integrated traffic, the current most popular application of ATM is data communication where ATM networks are used as backbones to interconnect legacy Local Area Networks (LAN) and servers, and support legacy applications such as FTP, e-mail, and Web browsing. Such legacy applications operate on TCP/IP platform, which requires that ATM must support TCP/IP in the protocol stack. Because of the popularity of TCP/IP, even some multimedia applications are based upon TCP/IP/ATM platform. Therefore, TCP/IP and ATM will coexist and interoperate in the future.

When we run TCP/IP over ATM, TCP/IP packets are segmented at the ATM layer into fixed-size cells. When an ATM switch drops a cell because buffer overflows, the rest of the cells belonging to the same packet of the discarded cell will still be transmitted. After the cells arrive at the destination, the destination fails to reassemble the packet to which the lost cell belonged. TCP has a mechanism to request the source to retransmit the corrupted packet. Once one or more cells constituting a packet are lost, the whole packet will be retransmitted. Hence, the loss of one cell is amplified to the loss of a packet. And a portion of network resources are wasted to transmit the corrupted and useless packets. This phenomenon was observed by Romanow and Floyd [1]. And two schemes named Partial Packet Discard (PPD) and Early Packet Discard (EPD) were proposed in [1] to enhance the efficiency of TCP over ATM. In PPD, when a cell is dropped from a switch buffer, all but the last cell in the packet are discarded even if the buffer can accommodate them. The last cell of the packet is used by the destination to delimit packet boundaries. Thus, PPD prevents the remaining part of the damaged packet wasting network resources. However, some damaged part of the packet still survive and reach the destination. In EPD, when the

queue length of the switch reaches a threshold level, the newly arriving packets will not be accepted to the buffer, and the entire packet is discarded. In addition, once a cell of an accepted packet is lost due to buffer overflow, all subsequent cells that belong to the same packet are discarded as in PPD. Comparing PPD with EPD, the former only discards the tail of the corrupted packet, while the latter may discard whole packets that might not have been damaged due to buffer overflow besides the tails of corrupted packets. PPD may be viewed as a special case of EPD with threshold equal to the buffer capacity.

Reference [1] studied the performance of TCP connections over ATM networks via discrete-event simulation and the performance improvements accrued by PPD and EPD were compared. Because EPD does not discriminate the packets while it discards them, severe unfairness in throughput was observed by simulations[5]. In [6], the authors proposed per-VC accounting and per-VC queueing techniques to deal with the unfairness, and simulation results verified the validity of their techniques. Because of the complexity of analytic model of TCP, almost all the analyses of TCP over ATM with PPD and EPD have been based on discrete-event simulation.

In order to get the numerical solution to the performance of PPD and EPD, details of TCP dynamics were elided and PPD and EPD were studied by assuming the packet arrival to be a stochastic process in [2, 3, 7]. In [2], all traffic sources were assumed to be homogeneous in terms of traffic characteristics. Each source generated cells according to an Interrupted Bernoulli Process (IBP). The cells generated during one ON period of IBP constituted a packet, and the ON and OFF periods were exponentially distributed. PPD and EPD were compared based on numerical results. It was shown that the performance of PPD is better than that of EPD when the packet length is short, and the performance of EPD with the optimal threshold is better than that of PPD when the packet length is

long. In [7], the selective-repeat ARQ (Automatic Repeat reQuest), an error control scheme in the transport layer, was integrated into the analysis of EPD, and the total cell arrivals (including the newly generated packets and retransmitted packets) were assumed to form a Bernoulli process. In [3], only one traffic source was considered and the packet arrival process was assumed to be Poisson. A new performance measure, i.e., goodput, was adopted to better describe the network behavior. The analysis was extended to the exponentially distributed ON-OFF traffic sources as well. The main limitation of [3] is that since they assumed only one traffic source, no packet interleaving could occur.

Up to now, the performance of EPD under heterogeneous traffic environment has not been studied. In the real network environment, the user applications generate packets in a bursty manner, i.e., in the ON state, cells are generated and constitute a packet, while in the OFF state, no cells are generated. Since the ON and OFF periods are generally distributed, the traffic process is an ON-OFF semi-Markov process. In [8], the empirical distribution of TELNET packet interarrival times, i.e., the duration of OFF period, was found to be heavy-tailed (approximated by a Pareto distribution), and the exponential distribution resulted in seriously underestimating the burstiness of TELNET traffic. For heavy-tailed OFF period, the longer the source has sojourned in the OFF state, the longer it will further sojourn in the OFF state, while the exponential distribution exhibits memoryless property. Therefore, the IBP model in [2] does not exactly describe the behavior of the packet traffic in legacy applications. After the flow of packets enters the ATM network, the statistical multiplexing will smooth the burstiness of the traffic, and randomize the interarrival intervals of the cells. Therefore, at the center of the network, the packet arrival process may be approximated by Poisson. Considering an ATM switch in the network, its input traffic consists of the flows generated by the users far away from the switch, and those near the switch (including

those directly connected to the switch). Thus, such heterogeneous traffic is closer to the real environment.

In this paper, we study the performance of EPD with heterogeneous traffic sources. The sources include Poisson, and ON-OFF with long-tailed sojourn time distribution. The long-tailed distribution is approximated by a hyperexponential distribution. The fairness of EPD is investigated. Although it is not impossible to study the model under consideration by using the conventional methods in [2, 3, 7], i.e., constructing Markov equations by hand, and then writing a program to solve the equations, it is rather tedious and error-prone. In this paper, we use stochastic reward nets (SRNs) [12] to concisely describe the system model. The underlying Markov models are generated automatically from the stochastic reward net models and solved by SPNP (Stochastic Petri Net Package), a widely circulated package developed by us [13, 14].

The organization of the paper is as follows. In Section 2, a brief description of stochastic reward nets is given. In Section 3, the stochastic reward net model for EPD under heterogeneous traffic environment is presented. Numerical results obtained by using *SPNP* are discussed in Section 4. The fairness and the effect of time-varying available bandwidth are studied. Conclusions are drawn in Section 5.

## 2 A Brief Description of Stochastic Reward Nets

Stochastic reward nets are extensions of stochastic petri nets. We give a brief overview of these constructs before presenting the model of EPD with heterogeneous traffic sources.

A Petri net is a directed graph with two disjoint types of nodes: *places* and *transitions*. A directed arc connecting a place (transition) to a transition (place) is called an input (output) *arc* of the transition. A positive integer called multiplicity can be associated with

each arc. Places connected to a transition by input arcs are called the input places of this transition, and those connected by means of output arcs are called the output places. Each place may contain zero or more tokens. A transition is *enabled* if each of its input places has at least as many tokens as the multiplicity of the corresponding input arc. A transition can *fire* when it is enabled, and upon firing, a number of tokens equal to the multiplicity of the input arc is removed from each of input places, and a number of tokens equal to the multiplicity of the output arc is deposited in each of its output places. The state of a Petri net is characterized by a number of tokens in each place which is called its marking. The initial number of tokens assigned to the places in a Petri net determines the initial state (marking) of the net.

Stochastic Petri nets are Petri nets where exponentially distributed firing time is attached to each transition. In Generalized Stochastic Petri nets (GSPN), transitions are allowed to be either *timed* (exponentially distributed firing time) or *immediate* (zero firing time). A marking of a GSPN is called *vanishing* if at least one immediate transition is enabled in the marking and *tangible* otherwise. GSPN also introduces *inhibitor arc* connecting a place to a transition. A transition with an inhibitor arc can not fire if the input place of the inhibitor arc contains more tokens than the multiplicity of the arc.

Stochastic reward nets (SRNs) are based on GSPN but extend them further [12]. In SRN, every tangible marking can be assigned a reward rate. It can be shown that an SRN can be mapped into a Markov reward model. Therefore, by using software packages [13, 14], the underlying Markov reward model of an SRN can be automatically generated and solved. Thus a variety of performance measures can be specified and calculated using a very convenient formalism. SRN also allows several other features that makes specification convenient: (1) each transition may have an enabling function (also called a guard) so

that a transition is enabled only if its (marking dependent) enabling function is true, (2) marking dependent arc multiplicities are allowed, (3) transitions can also have priorities. To represent an SRN as a graph, places are represented by circles and tokens are black dots (or integers) inside places. Immediate transitions are drawn as thin bars and timed transitions as white boxes. Inhibitor arcs have a small hollow circle instead of arrows at their terminating ends. The current number of tokens in place  $p$  is denoted as  $\#p$ .

### 3 Stochastic Reward Net Model for Early Packet Discard

#### 3.1 Early Packet Discard

We assume that the buffer capacity is  $N$  (cells), and that the threshold of EPD algorithm is  $K$  (cells). From the description of EPD algorithm [5, 6] in Fig. 1, we can see that the packets arriving into the network face three eventualities: successfully transmitted, totally dropped when the first cell of the packet found the queue length exceeding the threshold, and partially discarded when some middle cell of the packet sees buffer overflow. We assume the total number of packets arriving during time period of duration  $t$  as  $P_{in}(t)$ , the number of successfully transmitted packets as  $P_{out}(t)$ , the number of totally dropped packets as  $P_1(t)$ , and the number of partially discarded packets as  $P_2(t)$ , then

$$P_{in}(t) = P_1(t) + P_2(t) + P_{out}(t) \quad (1)$$

Then the effective throughput  $TH$  defined in [1] is:

$$TH = \lim_{t \rightarrow \infty} \frac{P_{out}(t)}{P_2(t) + P_{out}(t)} \quad (2)$$

The goodput  $G$  as defined in [3] is:

$$G = \lim_{t \rightarrow \infty} \frac{P_{out}(t)}{P_{in}(t)} \quad (3)$$



```

when a cell arrives at an ATM switch :
if ( this is the first cell of the packet) then
{
  if (queue length >= K) then discard the cell;
  else accept the cell into the buffer;
}
else
{
  if (any cell of the packet has been discarded) then discard the cell;
  else
  {
    if (queue length =N) then discard the cell;
    else accept the cell into the buffer;
  }
}

```

Figure 1: The EPD Algorithm

In order to get the  $TH$  and  $G$ , a series of complicated derivations were introduced in [3, 7]. We will show in the following subsections that  $TH$  and  $G$  can be easily obtained by using SRN.

### 3.2 SRN Model for EPD with Single Poisson Source

In this simple case, we assume the cell arrival process is Poisson and the length of packet is geometrically distributed in terms of cells. The cell arrival rate is  $\lambda$ . An arriving cell is the head of the packet with probability  $q$ . Then the mean packet length and packet arrival rate are  $\frac{1}{q}$  and  $\lambda q$ , respectively. The behavior of the buffer can be described with the SRN model shown in Fig. 2.

The firing of the immediate transition  $t_8$  represents the event that the arriving cell is the head of a packet. If it finds the queue length (the number of tokens in place *Queue*) is less than the threshold  $K$ , the immediate transition  $t_3$  fires and the cell enters place *Queue*. The cell transmission time is deterministic, which is approximated by a 3-stage Erlang distribution with rate  $3\mu$ . If the head of the packet finds the queue length exceeding the threshold  $K$ , then it will be dropped, the immediate transition  $t_1$  fires and one token is deposited in place

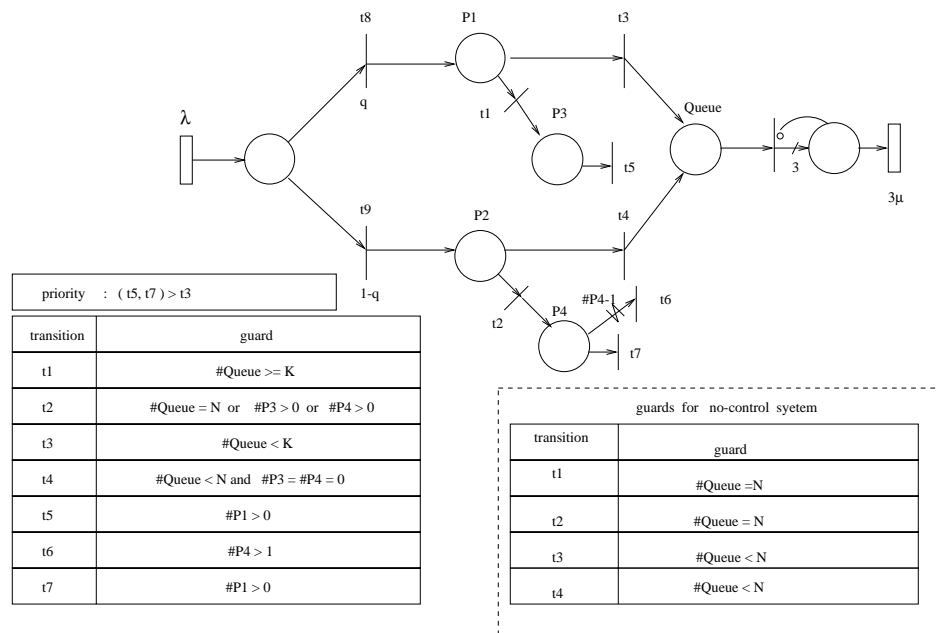


Figure 2: Stochastic Reward Net Model for EPD with Single Poisson Source

$P_3$  which is used to remember that the entire packet is discarded.

The firing of the immediate transition  $t_9$  represents the event that the arriving cell is not the head of a packet. If it finds buffer full, or the head of the packet has been discarded (i.e., there is one token in place  $P_3$ ), or one of its predecessor cell in the same packet has been discarded (i.e., there is one token in place  $P_4$ ), then the cell is discarded, otherwise it enters the buffer. If the cell is discarded, one token will be deposited in the place  $P_4$ . Immediate transition  $t_6$  and the variable cardinality of arc from  $P_4$  to  $t_6$  are used to guarantee the number of tokens in  $P_4$  be less than 2, which is used to reduce the number of states of the system.

Once a new packet's head enters place  $P_1$ , the tokens in the places  $P_3$  and  $P_4$  are cleared by immediate transitions  $t_5$  and  $t_7$ , which ensures that whether the current packet is discarded (damaged) or not has no relationship with the successive packet. The correct operation is guaranteed by giving  $t_5$  and  $t_7$  higher priority than  $t_3$ .

Obviously, any time we look at the state of the SRN in Fig. 2, a token in  $P_3$  represents that the packet is totally discarded, and a token in  $P_4$  represents that the packet is partially damaged. In other word, when the head of a packet arrives, the state of  $P_3$  and  $P_4$  gives the information about the previous packet's fate, successfully transmitted or totally discarded or partially damaged. SRN provides an index named *reward rate* to get the probabilities of the events we are interested in. Reward functions we use are shown in Fig. 3. According to the PASTA(Poisson Arrivals See Time Averages) theorem [15], the reward function *partial( )* gives the probability of a packet being partially damaged, and *total( )* presents the probability of a packet being totally discarded. Then the performance indices  $TH$  and  $G$  can be obtained from the reward functions.

```

double partial(){
    if ( mark (" P3 ")==0 && mark (" P4 ")>0)
        return(1.0);
    else
        return(0);
}
double total(){
    if ( mark (" P3 ")>0)
        return(1.0);
    else
        return(0);
}
double goodput(){
    if( mark("P3")==0 && mark ("P4")==0)
        return(1.0);
    else
        return(0);
}

```

Figure 3: The Reward Functions for the Poisson Model

$$TH = \frac{1 - \text{partial}() - \text{total}()}{1 - \text{partial}()} \quad (4)$$

$$G = 1 - \text{partial}() - \text{total}() \quad (5)$$

For SRN model of the queueing system without EPD or PPD control, we just need modify the guards of immediate transition  $t_1$ ,  $t_2$ ,  $t_3$  and  $t_4$  in Fig. 2 (see the table in the dashed rectangle). For the system without EPD or PPD control, we do not discriminate the packet is totally discarded or partially damaged. We define the reward function for goodput as *goodput()* (see Fig. 3).

### 3.3 SRN Model for EPD with Single ON-OFF Source

The model in Fig. 2 can be easily extended to the case where the cell arrivals are governed by an ON-OFF source [9]. During OFF period, no cells arrive, while during ON period, cells are generated according to Poisson process with rate  $\delta$ . The cells generated during one ON period constitute a packet.

Recent studies have shown that the traffic in the Internet can not be described by ON-OFF model with exponentially distributed durations [4, 8, 10]. In [8], the empirical distribution of TELNET packet interarrival times, i.e., the duration of OFF period, was found to be heavy-tailed, being well described by distributions such as the Pareto, and the exponential distribution resulted in seriously underestimating the burstiness of TELNET traffic. For heavy-tailed OFF period, its complementary cumulative distribution function decays more slowly than that of the exponential. And the longer the source has sojourned in the OFF state, the longer it will sojourn in the OFF state further, while the exponential distribution exhibits memoryless property. Although it is known that heavy-tailed distributions can have a dramatic effect upon performance, it is often difficult to describe this effect in detail because performance models which account for heavy-tailed distributions tend to be difficult to analyze. In [11], authors presented an algorithm to approximate the Pareto distribution with hyperexponential distribution (a finite mixture of exponentials). With this approximation, the system with heavy-tailed distribution becomes tractable.

In this paper, we assume the ON period, which corresponds to the size of packets, is exponentially distributed with mean  $1/\alpha$ . The rationale behind this assumption is that although the file size in the Internet is proved to be heavy-tailed, after the files are segmented into packets, the packet size seems to depend on the type of application. For example, for the FTP application, the packet size is close to the value of the Maximal Transfer Unit (MTU, for Ethernet, it is about 1500 byte), for the TELNET application, it may be any value less than MTU. The exponential will make our model simpler.

For the OFF period, we assume it is hyperexponentially distributed, i.e., the cumulative distribution function  $F_{off}(x)$  is given by:

$$F_{off}(x) = \sum_{i=1}^m c_i (1 - e^{-\gamma_i x}) \quad (6)$$

where

$$\sum_{i=1}^m c_i = 1 \quad (7)$$

Then, the mean OFF duration is given by:

$$E[OFF] = \sum_{i=1}^m \frac{c_i}{\gamma_i} \quad (8)$$

The mean packet arrival rate  $R_p$ , the mean cell arrival rate  $R_c$  and the mean packet length  $E_l$  are

$$R_p = \frac{\alpha}{1 + \alpha E[OFF]} \quad (9)$$

$$R_c = \frac{\delta}{1 + \alpha E[OFF]} \quad (10)$$

$$E_l = \frac{R_c}{R_p} = \frac{\delta}{\alpha} \quad (11)$$

respectively.

The SRN model with ON-OFF source is illustrated Fig. 4, where the subnet in the dashed rectangle models a hyperexponential with parameters  $(c_i, \gamma_i)$  ( $i = 1, 2, \dots, m$ ). Now we show how to get the  $TH$  and  $G$  via reward functions in this case. Fig. 5 presents an example how the tokens in palces  $P3'$  and  $P4'$  change with the arrival of cells. The head of first packet shown in Fig. 5 is not discarded, however the buffer overflows at time  $x1$  and all the successive cells are discarded. The state of  $(\#P3', \#P4')$  is changed upon the arrival of next packet at time  $x2$ . The second packet in Fig. 5 is successfully transmitted. The head of the third packet in Fig. 5 is discarded because of the queue length exceeding the threshold. The loss of the head of the third packet occurs at time  $x3$ , and changes the state of  $(\#P3', \#P4')$  to  $(1, 0)$ . The second cell of the packet changes the state of  $(\#P3', \#P4')$  to  $(1, 1)$  at time  $x4$ , and the state is maintained till time  $x5$ . The fourth packet is successfully transmitted. The fifth packet only contains one cell, and the state of

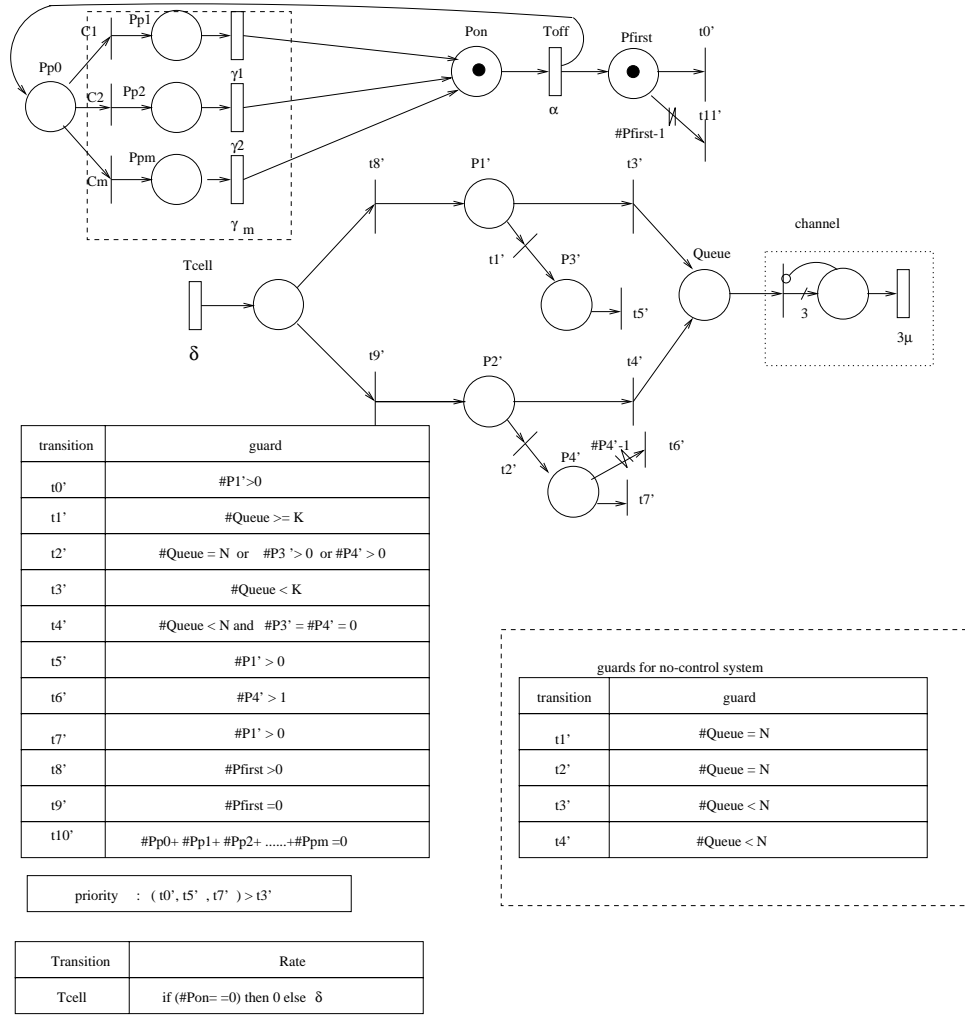


Figure 4: Stochastic Reward Net Model for EPD with Single ON-OFF Source

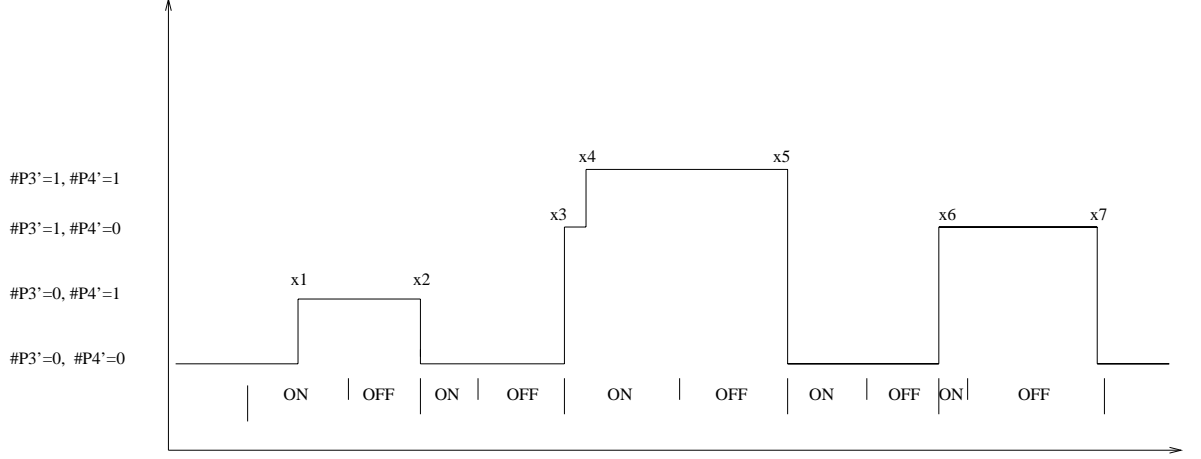


Figure 5: The State of  $(\#P3', \#P4')$  in the ON-OFF Traffic Case

$(\#P3', \#P4')$  is maintained at  $(1, 0)$  till time  $x7$ . Therefore, the head of a packet can detect if the previous packet is partially damaged or totally discarded or successfully transmitted according to the state of  $(\#P3', \#P4')$  at the instant of its arrival. Because this state is the same as that of  $(\#P3', \#P4')$  during the previous OFF period, we can calculate the reward function as shown in Fig. 6. For SRN model of the queueing system without EPD or PPD control, we just need modify the guards of immediate transition  $t'_1$ ,  $t'_2$ ,  $t'_3$  and  $t'_4$  in Fig. 4 (see the table in the dashed rectangle). We define the reward function for packet loss as  $loss()$  (see Fig. 6). Then

$$G = 1 - loss() \quad (12)$$

### 3.4 SRN Model for EPD with Heterogeneous Sources

Now we present the SRN model for EPD with heterogeneous sources in Fig. 7. There are  $N_p$  Poisson sources and  $N_i$  ON-OFF sources (The black rectangle represents the component within the dashed rectangle in Fig. 4). For each source, the packets will be discarded only



```

double partial(){
    if ( mark ( " P3' " ) ==0 && mark ( " P4' " ) >0 && mark ( "Pon" ) ==0)
        return(1.0);
    else
        return(0);
}
double total(){
    if ( mark ( " P3' " ) >0 && mark ( "Pon" ) ==0 )
        return(1.0);
    else
        return(0);
}
double loss(){
    if ( ( mark ( " P3' " ) >0 || mark ( " P4' " ) >0 ) && mark ( " Pon " ) ==0)
        return(1.0);
    else
        return(0);
}

```

Figure 6: The Reward Functions for the ON-OFF Model

according to the buffer occupancy, and independently of the behavior of the other sources.

## 4 Numerical Results

### 4.1 Performance Comparison When The Bandwidth Is Fixed

In this subsection, we will compare the partial packet loss ratio  $partial()$ , total packet loss ratio  $total()$ , effective throughput  $TH$  and goodput  $G$  for the sources with different traffic characteristics. We fix the service rate and the buffer capacity in this subsection:  $\mu = 50$ ,  $N = 100$ .

#### 4.1.1 All Poisson Sources

We compare the  $partial()$ ,  $total()$ ,  $TH$  and  $G$  for the Poisson sources with different parameters. We assume that there are three Poisson sources and no ON-OFF sources, i.e.,  $N_p = 3$ ,  $N_i = 0$ .

##### *A. underloaded queue*

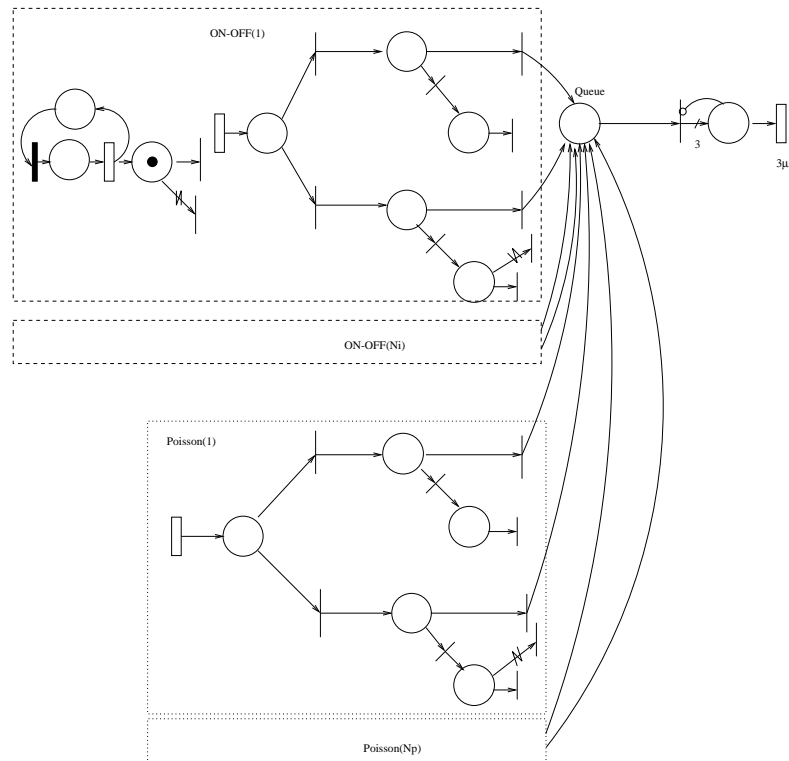


Figure 7: Stochastic Reward Net Model for EPD with Heterogeneous Sources ( $N_i$  ON-OFF and  $N_p$  Poisson)

First we study the performance of Poisson sources when the queue is underloaded.

Table 1 presents the performance of three Poisson sources with same packet arrival rates but different cell arrival rates and mean packet lengths. The parameters are chosen as:  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.06$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.1$ . It can be seen that:

- All the sources have the same *total()*.
- The source with the shortest packets has the least value of *partial()*, but the difference is insignificant.
- All the sources have almost the same *TH* and *G*, especially when the threshold of EPD approaches half the buffer capacity.
- *TH* is greater than *G*, but the difference is insignificant, and they have the same value when the threshold of EPD approaches half the buffer capacity.
- PPD works better than EPD. As the threshold of EPD decreases, the probability of a packet being partially discarded decreases but the probability of a packet being totally discarded increases. The amount of decrease in *partial()* is less than the increase in *total()*, which makes *TH* and *G* decrease as the threshold of EPD decreases.

Table 2 presents the performance of three Poisson sources with same cell arrival rates but different packet arrival rates and mean packet lengths. The parameters are chosen as:  $\lambda_1 = 15$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 15$ ,  $q_2 = 0.1$ ,  $\lambda_3 = 15$ ,  $q_3 = 0.4$ . All the observations for Table 1 can also be made in this case.

Table 3 presents the performance of three Poisson sources with same mean packet lengths but different cell arrival rates and packet arrival rates. The parameters are chosen as:  $\lambda_1 = 30$ ,  $q_1 = 0.05$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.05$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.05$ . All the observations for Table 1 can also be made except the second one should be modified as: The source with the highest packet arrival rate and cell arrival rate has the least value of *partial()*, but the difference

is insignificant. The reason behind this phenomenon is: higher cell arrival rate means the packet can enter the buffer quicker and the probability of being partially damaged decreases.

Table 4 presents the performance of three Poisson sources with different cell arrival rates, packet arrival rates and mean packet lengths. The parameters are chosen as:  $\lambda_1 = 30$ ,  $q_1 = 0.1$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.06$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.02$ . All the observations for Table 1 can also be made.

All the numerical results show that the sources with different traffic characteristics almost have the same goodput when the queue is underloaded with Poisson sources.

Table 5 presents the performance of three Poisson sources without EPD or PPD control. It shows that PPD will increase the goodput, however the increase is insignificant. The performance of EPD with small threshold is even worse than the system without control. Therefore, in an underloaded system with Poisson traffic sources, the benefit of EPD or PPD is not obvious.

### *B. overloaded queue*

Next we study the performance of Poisson sources when the queue is overloaded.

Table 6 presents the performance of three Poisson sources with same packet arrival rates but different cell arrival rates and mean packet lengths. The parameters are chosen as:  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.03$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.1$ . It can be seen that:

- All the sources have the same *total()*.
- The source with the shortest packets has the least value of *partial()*, and the difference *is significant*. The difference becomes insignificant as the threshold of EPD approaches half the buffer capacity.
- The source with the shortest packets has the highest goodput and effective throughput. But the goodput (effective throughput) of all the sources converge to the same value

when the threshold of EPD approaches half the buffer capacity.

- $TH$  is greater than  $G$ , and the difference *is significant*. But they converge to the same value when the threshold of EPD approaches half the buffer capacity.
- For the source with shorter packets (i.e., Sources 2 and 3), PPD outperforms EPD. But for the source with long packets, it is a little complicated. As the threshold of EPD decreases, the probability of a packet being partially discarded decreases but the probability of a packet being totally discarded increases. The amount of decrease in *partial()* is greater than the increase in *total()* when  $K = 90$  and  $K = 80$ . The amount of decrease in *partial()* is less than the increase in *total()* when  $K < 80$ . This means EPD with deliberately selected threshold outperforms PPD when the packet length is large. However, if we use  $TH$  as the performance index, we will get a false conclusion that “PPD outperforms EPD for all the sources”. In this sense,  $G$  is a better performance index than  $TH$ .

Table 7 presents the performance of three Poisson sources with same cell arrival rates but different packet arrival rates and mean packet lengths. The parameters are chosen as:  $\lambda_1 = 20$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.1$ ,  $\lambda_3 = 20$ ,  $q_3 = 0.4$ . All the observations for Table 6 can also be made.

Table 8 presents the performance of three Poisson sources with same mean packet lengths but different cell arrival rates and packet arrival rates. The parameters are chosen as:  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.02$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.02$ . It can be seen that:

- All the sources have the same *total()*.
- The source with the highest packet arrival rate and cell arrival rate has the least value of *partial()*. The difference between them becomes insignificant as the threshold of EPD decreases.

- The source with the highest packet arrival rate and cell arrival rate has the highest goodput or effective throughput. But the goodput (effective throughput) of all the sources converge to the same value when the threshold of EPD approaches half the buffer capacity.
- $TH$  is greater than  $G$ , and the difference between them *is significant*. But they converge to the same value when the threshold of EPD approaches half the buffer capacity.
- For source 1, PPD outperforms EPD. But for the sources 2 and 3, it is a little complicated. As the threshold of EPD decreases, the probability of a packet being partially discarded decreases but the probability of a packet being totally discarded increases. The amount of decrease in *partial()* is greater than the increase in *total()* when the threshold is high. The amount of decrease in *partial()* is less than the increase in *total()* when the threshold is low.

It can be concluded that the Poisson sources with different traffic characteristics obtain different goodput when PPD algorithm is adopted in the overloaded queue. And they obtain different goodput when EPD algorithm is adopted, unless the threshold is small enough. However, the unfairness among the sources, i.e., the difference in the goodput among the sources, is not very serious in the *All Poisson* case.

Fig. 8 illustrates the buffer occupancy while no control is adopted, PPD is enabled and EPD is enabled with threshold  $K = 80, 60, 40$  respectively. It shows that PPD and EPD make the buffer less congested.

Table 9 presents the performance of three Poisson sources without EPD or PPD control. It shows that both PPD and EPD will increase the goodput significantly. Therefore, in a overloaded system with Poisson traffic sources, the benefit of EPD or PPD is obvious.

Table 1: Performance Comparison of Three Poisson Sources With Same Packet Arrival Rate

(Underload),  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.06$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.1$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	2.539442e-06	5.135335e-07	6.375239e-08	7.591721e-09	8.956025e-10	1.055236e-10	2.521262e-14
total(1)	7.610176e-08	3.070637e-06	1.143146e-05	3.971285e-05	1.376547e-04	4.779398e-04	0.08733807
TH(1)	0.99999992	0.99999692	0.99998856	0.99996028	0.99986234	0.99952206	0.91266192
G(1)	0.99999738	0.99999641	0.99998851	0.99996027	0.99986234	0.99952206	0.91266192
partial(2)	9.808351e-07	1.841719e-07	2.273384e-08	2.706214e-09	3.192538e-10	3.761613e-11	8.987615e-15
total(2)	7.610176e-08	3.070637e-06	1.143146e-05	3.971281e-05	1.376547e-04	4.779398e-04	0.08733807
TH(2)	0.99999992	0.99999692	0.99998856	0.99996028	0.99986234	0.99952206	0.91266192
G(2)	0.99999894	0.99999674	0.99998854	0.99996028	0.99986234	0.99952206	0.91266192
partial(3)	5.996765e-07	1.045148e-07	1.241078e-08	1.457696e-09	1.712163e-10	2.014531e-11	4.8101711e-15
total(3)	7.610176e-08	3.070637e-06	1.143146e-05	3.97128e-05	1.376547e-04	4.779398e-04	0.08733807
TH(3)	0.99999992	0.99999693	0.99998856	0.99996028	0.99986234	0.99952206	0.91266192
G(3)	0.99999932	0.99999682	0.99998855	0.99996028	0.99986234	0.99952206	0.91266192

Table 2: Performance Comparison of Three Poisson Sources With Same Cell Arrival Rate

(Underload),  $\lambda_1 = 15$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 15$ ,  $q_2 = 0.1$ ,  $\lambda_3 = 15$ ,  $q_3 = 0.4$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=10
partial(1)	2.040275e-07	1.721589e-09	5.776004e-12	1.805626e-14	5.56218471217e-17	3.917407e-27
total(1)	5.585560e-09	1.320306e-07	6.336734e-07	3.036092e-06	1.45476520046e-05	0.044222194
TH(1)	0.999999944	0.99999867	0.999993663	0.999969639	0.99998545234	0.9557778054
G(1)	0.999997903	0.999998662	0.999993663	0.999969639	0.999985452348	0.9557778054
partial(2)	3.804765e-08	3.032477e-10	1.018492e-12	3.184641e-15	9.81071264149e-18	5.618908e-28
total(2)	5.585560e-09	1.320306e-07	6.336734e-07	3.036092e-06	1.45476520046e-05	0.044222194
TH(2)	0.999999944	0.999998679	0.999993663	0.999969639	0.999985452348	0.9557778054
G(2)	0.999999563	0.999998676	0.999993663	0.999969639	0.999985452348	0.9557778054
partial(3)	6.543854e-09	3.080644e-11	9.454424e-14	2.899982e-16	8.89578071066e-19	2.278721e-29
total(3)	5.585560e-09	1.320306e-07	6.336734e-07	3.036092e-06	1.45476520046e-05	0.044222194
TH(3)	0.999999944	0.999998679	0.999993663	0.999969639	0.999985452348	0.9557778054
G(3)	0.999999878	0.999998679	0.999993663	0.999969639	0.999985452348	0.9557778054

Table 3: Performance Comparison of Three Poisson Sources With Same Packet Length  
(Underload),  $\lambda_1 = 30$ ,  $q_1 = 0.05$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.05$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.05$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	1.077147e-06	1.257719e-07	1.022272e-08	8.178139e-10	6.528196e-11	5.217147e-12	2.783749e-16
total(1)	8.297817e-08	2.543419e-06	9.146435e-06	3.171164e-05	1.099076e-04	3.815895e-04	0.073516
TH(1)	0.99999991	0.99999745	0.99999085	0.99996828	0.99989009	0.99961841	0.9264838
G(1)	0.99999883	0.99999733	0.99999084	0.99996828	0.99989009	0.99961841	0.9264838
partial(2)	1.288276e-06	1.463824e-07	1.185148e-08	9.478335e-10	7.565891e-11	6.046428e-12	3.226232e-16
total(2)	8.297816e-08	2.543419e-06	9.146435e-06	3.171164e-05	1.099076e-04	3.815895e-04	0.073516
TH(2)	0.99999991	0.99999745	0.99999085	0.99996828	0.99989009	0.99961841	0.9264838
G(2)	0.99999862	0.99999731	0.99999084	0.99996828	0.99989009	0.99961841	0.9264838
partial(3)	1.368931e-06	1.543888e-07	1.238110e-08	9.872766e-10	7.874697e-11	6.292015e-12	3.357063e-16
total(3)	8.297817e-08	2.543419e-06	9.146435e-06	3.171164e-05	1.099076e-04	3.815895e-04	0.073516
TH(3)	0.99999991	0.99999745	0.99999085	0.99996828	0.99989009	0.99961841	0.9264838
G(3)	0.99999854	0.99999730	0.99999084	0.99996828	0.99989009	0.99961841	0.9264838

Table 4: Performance Comparison of Three Poisson Sources (Underload),  $\lambda_1 = 30$ ,  $q_1 = 0.1$ ,  
 $\lambda_2 = 10$ ,  $q_2 = 0.06$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.02$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	5.752077e-07	2.450823e-08	7.464434e-10	2.258006e-11	6.828599e-13	2.067842e-14	2.328455e-20
total(1)	9.320415e-08	2.077776e-06	7.285664e-06	2.523819e-05	8.745546e-05	3.034901e-04	0.05974708
TH(1)	0.99999991	0.99999792	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291
G(1)	0.99999933	0.99999789	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291
partial(2)	1.182622e-06	5.085506e-08	1.543707e-09	4.668576e-11	1.411831e-12	4.275311e-14	4.814131e-20
total(2)	9.320415e-08	2.077776e-06	7.285664e-06	2.523819e-05	8.745546e-05	3.034901e-4	0.05974708
TH(2)	0.99999991	0.99999792	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291
G(2)	0.99999872	0.99999787	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291
partial(3)	3.915558e-06	1.746529e-07	5.302990e-09	1.603174e-10	4.847669e-12	1.467938e-13	1.652933e-19
total(3)	9.320415e-08	2.077776e-06	7.2856e-06	2.523819e-05	8.745546e-05	3.034901e-04	0.05974708
TH(3)	0.99999991	0.99999792	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291
G(3)	0.99999599	0.99999774	0.99999271	0.99997476	0.99991254	0.99969651	0.94025291



Table 5: Performance of The System Without Control(Underload)

	Table 1	Table 2	Table 3	Table 4
G(1)	0.99999581	0.9999997	0.99999781	0.99999862
G(2)	0.99999725	0.9999999	0.99999681	0.99999725
G(3)	0.99999796	0.9999999	0.99999628	0.99999155

Table 6: Performance Comparison of Three Poisson Sources with Same Packet Arrival Rate (Overload),  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.03$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.1$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	0.14062011	0.04334734	0.00929997	0.00188741	0.00037637	7.485278e-05	1.512717e-07
total(1)	0.00445752	0.08373984	0.11289329	0.12258680	0.12928466	0.13736011	0.25566919
TH(1)	0.99481309	0.91246577	0.88604694	0.87718138	0.87066666	0.86262971	0.74433076
G(1)	0.85492236	0.87291280	0.87780673	0.87552578	0.87033896	0.86256513	0.74433065
partial(2)	0.10106446	0.03036747	0.00651336	0.00132196	0.00026362	5.2431937e-05	1.059663e-07
total(2)	0.00445752	0.08373984	0.11289329	0.12258680	0.12928466	0.13736001	0.25566919
TH(2)	0.99504133	0.91363754	0.88636657	0.87725092	0.87068124	0.86263278	0.74433076
G(2)	0.89447801	0.88589267	0.88059334	0.87609122	0.87045171	0.86258755	0.74433065
partial(3)	0.03423639	0.00835613	0.00156189	0.00029102	5.485362e-05	1.050068e-05	1.059663e-07
total(3)	0.00445752	0.08373984	0.11289329	0.12258680	0.12928466	0.13736001	0.25566919
TH(3)	0.99538445	0.91555451	0.88693010	0.87737750	0.87070824	0.86263854	0.74433077
G(3)	0.96130608	0.90790401	0.88554481	0.87712217	0.87066048	0.86262948	0.74433069

Table 7: Performance Comparison of Three Poisson Sources With Same Cell Arrival Rate

(Overload) ,  $\lambda_1 = 20$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.1$ ,  $\lambda_3 = 20$ ,  $q_3 = 0.4$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	0.38120115	0.0142188	0.00011018	7.099985e-07	1.656373e-09	2.407053e-11	2.437337e-20
total(1)	0.01272115	0.16272264	0.16889856	0.16995305	0.17115581	0.17361426	0.22289187
TH(1)	0.97944218	0.83493026	0.83108282	0.83004682	0.82884420	0.82638574	0.77710812
G(1)	0.60607768	0.82305847	0.83099124	0.83004623	0.82884418	0.82638573	0.77710812
partial(2)	0.07568435	0.00245259	1.911146e-05	1.234171e-07	8.906737e-08	4.192218e-12	4.250543e-21
total(2)	0.01272115	0.16272262	0.16889856	0.16995305	0.17156178	0.17361426	0.22289187
TH(2)	0.98623721	0.83687729	0.83109820	0.83004692	0.82843842	0.82638573	0.77710812
G(2)	0.91159449	0.83482477	0.83108232	0.83004682	0.82843821	0.82638573	0.77710812
partial(3)	0.01367154	0.00016055	8.12964703	4.059491e-09	1.010946e-12	1.021572e-13	7.902046e-23
total(3)	0.01272115	0.16272262	0.16889856	0.16995305	0.17124985	0.17361426	0.22289187
TH(3)	0.98710255	0.83725123	0.83110129	0.83004694	0.17124985	0.82638573	0.77710812
G(3)	0.97360730	0.83711681	0.83110062	0.83004694	0.17124985	0.82638573	0.77710812

Table 8: Performance Comparison of Three Poisson Sources With Same Packet Length

(Overload),  $\lambda_1 = 30$ ,  $q_1 = 0.02$ ,  $\lambda_2 = 20$ ,  $q_2 = 0.02$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.02$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=10
partial(1)	0.113636	0.045966	0.014703	0.004618	0.001458	0.000465	6.11e-06
total(1)	0.003587	0.074782	0.109523	0.124266	0.133743	0.143454	0.263813
TH(1)	0.995952	0.921614	0.888841	0.875156	0.866061	0.856478	0.736184
G(1)	0.882776	0.879251	0.875772	0.871114	0.864797	0.856079	0.736179
partial(2)	0.123266	0.049561	0.015847	0.004977	0.001571	0.000501	6.59e-06
total(2)	0.003587	0.074782	0.109523	0.124266	0.133743	0.143454	0.263813
TH(2)	0.995908	0.921318	0.888712	0.875111	0.866045	0.856473	0.736184
G(2)	0.873146	0.875657	0.874628	0.870755	0.864684	0.856043	0.736179
partial(3)	0.146071	0.058114	0.018294	0.005701	0.001792	0.000571	7.48e-06
total(3)	0.003587	0.074782	0.109523	0.124266	0.133743	0.143454	0.263813
TH(3)	0.995799	0.920603	0.888435	0.875021	0.866015	0.856463	0.736184
G(3)	0.850341	0.867102	0.872181	0.870032	0.864463	0.855973	0.736178

Table 9: Performance of The System Without Control(Overload)

	Table 6	Table 7	Table 8
G(1)	0.323143	0.156577	0.323143
G(2)	0.357934	0.438822	0.283012
G(3)	0.527959	0.708280	0.199124

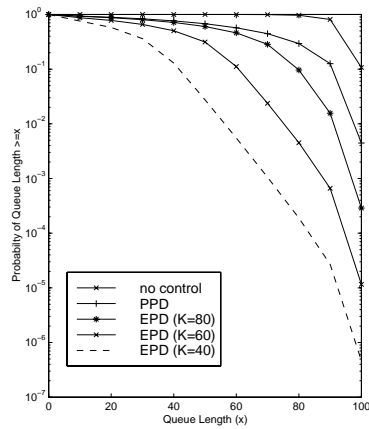


Figure 8: Buffer Occupancy Probability(No-Control, PPD and EPD)

### 4.1.2 Heterogeneous Sources

In this section, We will compare performance of heterogeneous sources. One source is Poisson, the other is ON-OFF with exponential ON duration and hyperexponential OFF duration. The hyperexponential distribution has three phases. The parameters are given in Table 10. In Case *I*, the Poisson source has the same mean packet arrival rate ( $R_p = 0.4$ ), mean cell arrival rate ( $R_c = 20$ ) and mean packet length ( $E_l = 50$ ) as the ON-OFF sources. In Case *II*,  $R_p = 4$ ,  $R_c = 20$ ,  $E_l = 5$ . In Case *III*,  $R_p = 0.8$ ,  $R_c = 40$ ,  $E_l = 50$ . In Case *IV*,  $R_p = 8$ ,  $R_c = 40$ ,  $E_l = 5$ . The goodput of the heterogeneous sources are compared in Table 11. It shows that:

- Both PPD and EPD will increase the goodput significantly. Therefore, in a system with ON-OFF traffic sources, the benefit of EPD or PPD is obvious.
- When the system is underloaded and packets are short, the difference between the goodputs of the heterogeneous sources is not very significant. Of course, it is more significant than that of the *All Poisson* case.
- When the system is overloaded or packets are long, the ON-OFF source has higher goodput than the Poisson source. The difference between the goodputs of the heterogeneous sources *is very significant*. And EPD algorithm makes this unfairness more serious. Because the burstiness of a source will be alleviated by the statistical multiplexing of ATM switches, the ON-OFF source may be viewed as the source that just enters the network, and the Poisson source may be viewed as the source being far away from the network node we are considering and has been regulated and smoothed by the switches it traversed. Therefore the queueing system with EPD algorithm gives higher goodput to the sources near the system. And the sources far from the system that have traversed several switches will have lower goodput.

Table 10: Performance Comparison of Heterogeneous Sources

	$\delta$	$\gamma_1$	$\gamma_2$	$\gamma_3$	$c_1$	$c_2$	$c_3$	$\alpha$	$\lambda$	$q$
I	50	3	0.45	0.01	0.9	0.09	0.01	1	20	0.02
II	50	30	4.5	0.1	0.9	0.09	0.01	10	20	0.2
III	100	6	0.9	0.02	0.9	0.09	0.01	2	40	0.02
IV	100	60	9	0.2	0.9	0.09	0.01	20	40	0.2

Table 11: Performance Comparison of Heterogeneous Sources

	no control	K=100(PPD)	K=90	K=80	K=70	K=60	K=50
I: G(1)	0.810817	0.924506	0.931494	0.933533	0.931454	0.926908	0.920259
I: G(2)	0.759460	0.935179	0.919368	0.908641	0.899701	0.890072	0.878466
II: G(1)	0.970737	0.984537	0.986366	0.984421	0.982022	0.979144	0.975595
II: G(2)	0.969075	0.987109	0.979431	0.976194	0.972521	0.972521	0.962698
III:G(1)	0.468232	0.816201	0.687529	0.740587	0.688921	0.687888	0.685997
III:G(2)	0.466393	0.506599	0.672722	0.430257	0.663329	0.659611	0.655352
IV: G(1)	0.586539	0.877712	0.725726	0.726108	0.724468	0.722297	0.719671
IV: G(2)	0.567318	0.790839	0.684101	0.679171	0.676731	0.674138	0.671051

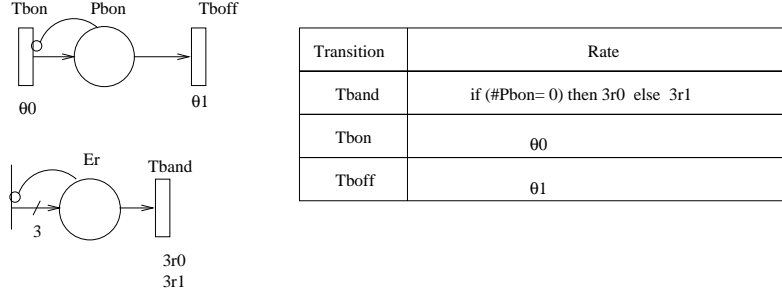


Figure 9: Stochastic Reward Net Model for Time-Varying Available Bandwidth

## 4.2 The Effect of Time-Varying Available Bandwidth

The EPD algorithm is mainly used to enhance the efficiency of transferring IP packets over ATM networks. Note that the ATM switches will support integrated services. Since the Constant-Bit-Rate services and real-time Variable-Bit-Rate services have higher priority over data services, the bandwidth available to the IP traffic is time-varying. In this subsection, we will assume the bandwidth available to the IP traffic is characterized by an ON-OFF model. The duration of ON (OFF) period is exponentially distributed with mean value  $1/\theta_0$  ( $1/\theta_1$ ), and the bandwidth available to the IP traffic during ON (OFF) period is  $r_0$  ( $r_1$ ). Although this ON-OFF time-varying bandwidth is not very close to the real case, it is sufficient for the purpose of studying the effect of time-varying available bandwidth on the performance of EPD. The SRN model for the time-varying service rate is illustrated in Fig. 9. The performance of EPD with time-varying available bandwidth may be studied by substituting the channel model in Fig. 7 (3-stage Erlang) with that in Fig. 9.

Table 12: Performance Comparison of Three Poisson Sources,  $\lambda_1 = 30$ ,  $q_1 = 0.1$ ,  $\lambda_2 = 10$ ,  $q_2 = 0.01$ ,  $\lambda_3 = 6$ ,  $q_3 = 0.02$ ,  $\theta_0 = 0.2$ ,  $\theta_1 = 0.5$ ,  $r_0 = 27$ ,  $r_1 = 40$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=40
G(1)	0.850982	0.727582	0.676247	0.664681	0.662118	0.660963	0.659387
G(2)	0.234096	0.491096	0.624951	0.654943	0.660288	0.660617	0.659322
G(3)	0.429889	0.620386	0.653858	0.660308	0.661273	0.660801	0.659356

Table 13: Performance Comparison of Heterogeneous Sources,  $\delta = 50$ ,  $\gamma_1 = 30$ ,  $\gamma_2 = 4.5$ ,  $\gamma_3 = 0.1$ ,  $c_1 = 0.9$ ,  $c_2 = 0.09$ ,  $c_3 = 0.01$ ,  $\alpha = 10$ ,  $\lambda = 20$ ,  $q = 0.2$ ,  $\theta_0 = 0.2$ ,  $\theta_1 = 0.5$ ,  $r_0 = 27$ ,  $r_1 = 40$

	K=100(PPD)	K=90	K=80	K=70	K=60	K=50	K=40
G(1)	0.870101	0.795985	0.794526	0.791457	0.926908	0.784088	0.779784
G(2)	0.788379	0.757906	0.752052	0.748001	0.890072	0.739048	0.733827

#### 4.2.1 All Poisson Sources

From Table 12, it is seen that PPD is very unfair in terms of goodput and EPD is approximately fair in terms of goodput if we adjust the threshold appropriately.

#### 4.2.2 Heterogeneous Sources

In this section, we assume that the packets are very short:  $E_l = 5$ . The goodputs of heterogeneous sources are given in Table 13. Even under this condition, both EPD and PPD display unfairness. And the ON-OFF sources have higher goodput.

## 5 Conclusions

In this paper, we constructed a performance model of PPD and EPD in ATM switches. We studied the performance of EPD with heterogeneous traffic sources. The sources in-

cluded Poisson, and ON-OFF with long-tailed sojourn time distribution. The long-tailed distribution is approximated by a hyperexponential distribution. The fairness of EPD is investigated. It was found from our numerical results that the ON-OFF source got higher goodput than the Poisson sources and in All Poisson case PPD and EPD provide nearly fair service to the sources. Because the burstiness of a source will be alleviated by the statistical multiplexing of ATM switches, the ON-OFF source may be viewed as the source that just enters the network, and the Poisson source may be viewed as the source being far from the network node we are considering and has been regulated and smoothed by the switches it traversed. Therefore the queueing system with EPD algorithm gives higher goodput to the sources near the system. And the sources far from the system that traversed several switches will have lower goodput. Some of the per VC-based schemes [6] may be adopted to alleviate the unfairness, however, these schemes make the ATM switches rather complex. Because at the center of a large ATM network, the traffic of individual flow has been regulated and smoothed by the switches it traversed, these sources may be modeled as Poisson process. Our studies have shown that PPD and EPD is approximately fair in the *All Poisson* case. Therefore, no per VC-based schemes is needed at the core of large ATM networks. These schemes may be used at the edge of the ATM cloud. Such configuration can make the core of the ATM networks work at high speed.

## References

- [1] A. Romanow, S. Floyd, "Dynamics of TCP Traffic over ATM Networks," *IEEE J -SAC*, pp. 633-641, May 1995.
- [2] K. Kawahara, K. Kitajima, T. Takine, Y. Oie, "Packet Loss Performance of Selective



- Cell Discard Schemes in ATM Switches,” *IEEE J-SAC*, pp. 903-912, June 1997.
- [3] Y. Lapid, R. Rom, M. Sidi, “Analysis of Discard Policies in High-Speed Networks,” *IEEE J-SAC*, pp. 764-777, June 1998
- [4] A. Adas, “Traffic Model in Broadband Networks,” *IEEE Communication Magazine*, pp. 82-89, Aug. 1997.
- [5] K. Y. Sui, H. Y. Tzeng, “Performance of TCP over ATM with Time-Varing Available Bandwidth,” *Computer Communications*, v.19, n.11, Sep. 1996.
- [6] H. Y. Li, K .Y. Siu, H. Y. Tzeng, C. Ikeda, H. Suzuki, “On TCP Performance in ATM Networks with Per-VC Early Packet Discard Mechanisms,” *Computer Communications*, v.19, n.13, Nov 1996.
- [7] Naoki. Tsukutani, Kenji. Kawahara,Tetsuya. Takine,Hideki. Sunahara,Yuji. Oie, “Throughput Analysis of Selective Cell Discard Schemes in Transport Layer over ATM Networks,” *GLOBECOM'97*, pp. 1067-1074.
- [8] V. Paxson, S. Floyd, “Wide Area Traffic: The Failure of Poisson Modeling,” *IEEE/ACM Trans. on Networking*, pp. 227-444, June 1995.
- [9] R. Jain, S. A. Routhier, “Packet Train: Measurements and A New Model for Computer Network Traffic”, *IEEE J-SAC*, pp. 986-995, 1986.
- [10] M. E. Crovella, A. Bestavros, “Self-similarity in World Wide Web Traffic- Evidence and Possible Causes”, *Sigcomm'96*, pp. 160-169.
- [11] A. Feldmann, W.Whitt, “ Fitting Mixtures of Exponentials To Long-Tail Distributions To Analyze Networks,” *Performance Evaluation*, pp. 245-280, Jan. 1998.

- [12] G. Ciardo, J. Muppala, K. S. Trivedi, “Analyzing Concurrent and Fault-tolerant Software Using Stochastic Reward Nets”, *Journal of Parallel and Distributed Computing*, 15, pp. 255-269, 1992
- [13] G. Ciardo, J. Muppala and K. Trivedi. “SPNP: Stochastic Petri Net Package,” *International Conference on Petri Nets and Performance Models*, Kyoto, Japan, December 1989.
- [14] G. Ciardo, K. S. Trivedi, “Manual for SPNP: Stochastic Petri Net Package”, version 5.0, CACC, ECE Department, Duke University, 1996.
- [15] R. W. Wolff, *Stochastic Modeling and the Theory of Queues*, Prentice Hall, 1989.