Performance Analysis of Error Recovery Schemes in High Speed Network - Part II

Arne A. Nilsson

Fu-Yung Lai

Center for Communications and Signal Processing
Electrical and Computer Engineering Department
North Carolina State University

CCSP TR-89/19

Contents

1 Introduction **2**

2 The Network Model **3**

3 End-to-end Error Control Scheme **5**

3.1 Link level model 5

3.2 Blocking Probability 8

3.2.1 Computation of P_{B_M} 8

3.2.2 Approximate Solution of the Blocking Probability,P_{B_i} 11

4 Link-by-link Error Control **19**

4.1 Link level model 19

4.2 Blocking Probability 21

5 Results and Conclusion **22**

1 Introduction

This paper considers the performance of the link-by-link and end-to-end error recovery schemes which are usually performed in the high speed communication environment. In the link-by-link error recovery approach, two adjacent nodes in a path locally detect and recover from packet error. In the end-to-end approach, error recovery is done solely on the basis of a single end-to-end protocol.

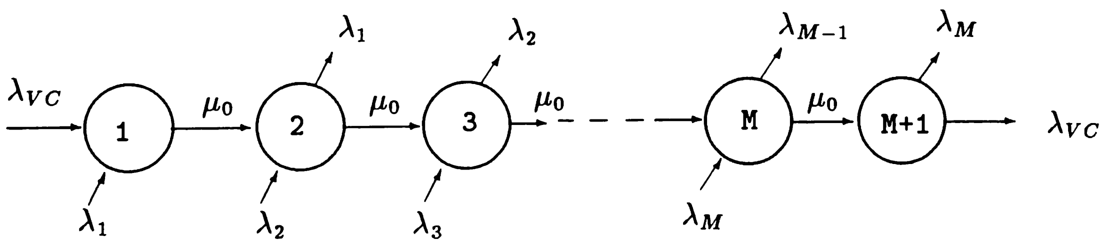
In this paper, we propose analytic performance models for comparing the performance of the link-by-link with end-to-end approaches with respect to error control. The models consider the effects of packet error rate, propagation delays of messages, separate virtual circuit finite buffers, and timeout mechanisms. A result of this paper is that when the packet error rate on a link is low, the end-to-end error recovery scheme can achieve shorter delay.

This paper is organized as follows. In section 2, the network model under study is described. The performance model of the end-to-end approach is given in section 3. The computation of the approximate blocking probability is given in section 3.2. The model of the link-by-link approach is given in section 4. Finally, results and conclusion are given in section 5.

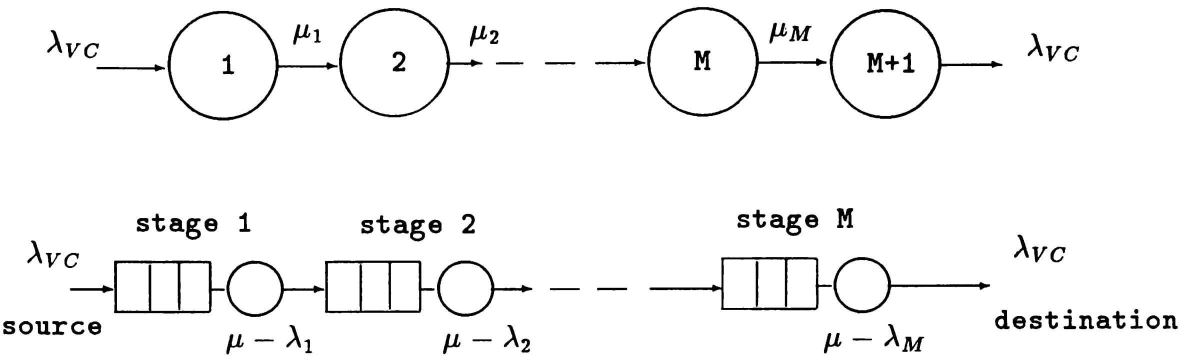
2 The Network Model

The model under study will be based on a virtual circuit channel in the communication network, see Fig(1). We assume that all but the first node will maintain a separate buffer for each virtual circuit and that there is enough buffer capacity for the first node to store all new packets. In addition to the traffic along the link of the virtual circuit, messages from other virtual circuits, which are called external traffic, also join the queues and obtain service.

The assumption made for the analysis is that the arrival process of external traffic at each node is independent of the traffic along the virtual circuit and independent from node to node. Let's denote the average arrival rate of the traffic along the link of the virtual circuit as λ_{VC} , the mean value of the message transmission rate of each link as μ_0, and the external traffics at node j as $\lambda_j, j = 1,2,\cdots,M$. We assume that the traffic along the virtual circuit and the external traffic are independent Poisson processes. The service time is assumed to be exponentially distributed at each node of the virtual circuit.



Fig(1) virtual circuit model (before capacity reduction)



Fig(2) virtual circuit model (after capacity reduction)

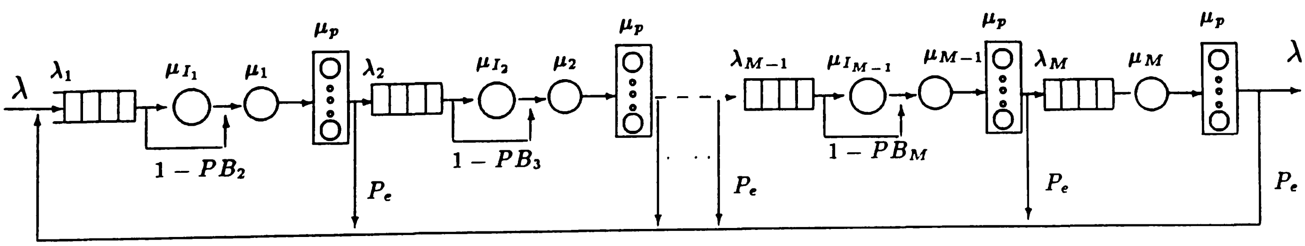
Since the external traffic will share the link capacity with the normal traffic λ_{VC}, this causes a reduction of the link capacity for transmitting the normal traffic along the virtual circuit. By capacity reduction[4], the virtual circuit model is shown in Fig(2), where $\mu_i, i = 1, 2, \cdots, M$, is the effective capacity of link i in the virtual circuit. For simplicity, we have assumed that the geographical distance between each node is the same. We assume that the propagation delay and the transmission time are of the same order. As in [1], we focus on the effect of the traffic ,λ_{VC}, entering a virtual circuit of M links which begins at node 1(source node) and ends at node M+1(destination node) to evaluate the performance of the error recovery schemes. The model in [3] ignores the packet error probability, and the model in [2] does not consider the finite buffer problem. In this model, we will consider the effect of packet error rate, finite buffer capacities, propagation delay of messages, timeout mechanism, and traffic load of the virtual circuit.

3 End-to-end Error Control Scheme

In an end-to-end error recovery approach, the source node of the virtual circuit first transmits a packet and keeps the packet until an acknowledgement of the packet from the destination node is received. The function of the intermediate node of the virtual circuit is very simple. If an error of a packet is detected by an intermediate node, the packet will be discarded immediately. If a packet is blocked due to the fact that the buffer is full in the next node, the packet will be held in the node and the link will sit idle until a receive ready signal is received. If an ACK message of a packet is not received during a specified timeout period after the packet is transmitted by the source node, the packet will be retransmitted from the source node. We assume that a timeout occurs only if the packet is in error.

3.1 Link level model

A link level virtual circuit model of the end-to-end error recovery scheme is shown in Fig(3).



Fig(3) Link level model of the end-to-end error recovery scheme

We assume that the traffic at each node is a Poisson process with rate λ_i , $i = 1, 2, \cdots, M$. For the idle time due to a packet being blocked and packet transmission time, we assume that they both are exponentially distributed with mean $1/\mu_{I_i}$ and $1/\mu_i$, $i = 1, \cdots, M$, respectively. We assume that the packet will not be blocked by the last node of the virtual circuit, thus $1/\mu_{I_M}$ is equal to 0. Let $1/\mu_p$ be the propagation delay of a packet through each link. With probability P_e, a packet may be in error when it passes through a link of the virtual circuit. Let K be the maximum buffer size of a node and $P_{B_i}, i = 2, 3, \cdots, M$, be the probability that the buffer size of node i is full. P_{B_1} is equal to 0, since we assume that the first node has enough buffer space to store all new packets. In steady state, we get

$$\frac{1}{\mu_{I_{M-1}}} = \frac{1}{\mu_M} + \frac{1}{\mu_p} \tag{1}$$

$$\frac{1}{\mu_{I_i}} = \frac{1}{\mu_{i+1}} + \frac{1}{\mu_p} + P_{B_{i+2}} \cdot \frac{1}{\mu_{I_{i+1}}} \qquad , i = 1, 2, \cdots, M-2. \tag{2}$$

$$\lambda_M = \frac{\lambda_{VC}}{1 - P_e} \tag{3}$$

$$\lambda_i = \frac{\lambda_{i+1}}{1 - P_e} \qquad , i = 1, 2, \cdots, M-1. \tag{4}$$

Let $T_i, i = 1, 2, \cdots, M$, be the sum of the system time of the transmission queue and the system time of the infinite server queue. To find T_i, we can view the transmission queue as an M/G/1 queue. Let $B_i^*(s)$ be the Laplace transform of the transmission time density function at node i. We have

$$B_M^*(s) = \frac{\mu_M}{s + \mu_M} \tag{5}$$

and

$$B_i^*(s) = \frac{\mu_i}{s+\mu_i} \cdot \left(1 - P_{B_{i+1}} + P_{B_{i+1}} \cdot \frac{\mu_{I_i}}{s+\mu_{I_i}}\right) \qquad ,i = 1,2,\cdots,M-1. \tag{6}$$

From the Pollaczek-Khinchin mean value formula[8], we get

$$T_i = \overline{x_i} + \frac{\lambda_i \overline{x_i^2}}{2(1-\rho_i)} + \frac{1}{\mu_p} \qquad ,i = 1,2,\cdots,M. \tag{7}$$

where

$$\begin{aligned} \rho_i &= \lambda_i \cdot \overline{x_i} \\ \overline{x_i} &= -\frac{d}{ds} B_i^*(s)\Big|_{s=0} \\ \overline{x_i^2} &= \frac{d^2}{ds^2} B_i^*(s)\Big|_{s=0} \end{aligned}$$

Thus, we get

$$\overline{x_M} = \frac{1}{\mu_M} \tag{8}$$

$$\overline{x_M^2} = \frac{2}{\mu_M^2} \tag{9}$$

$$\overline{x_i} = \frac{1}{\mu_i} + P_{B_{i+1}} \cdot \frac{1}{\mu_{I_i}} \qquad ,i = 1,2,\cdots,M-1. \tag{10}$$

$$\overline{x_i^2} = 2 \cdot \left(\mu_i^2 + P_{B_{i+1}} \cdot \frac{1}{\mu_{I_i}} \cdot \frac{1}{\mu_i} + P_{B_{i+1}} \cdot \frac{1}{\mu_{I_i}^2}\right) \qquad ,i = 1,2,\cdots,M-1. \tag{11}$$

Let P_{fail} be the probability that a transmission of a packet will not be successfully received by the destination node. We get

$$P_{fail} = 1 - (1 - P_e)^M.$$

Let N_r be the average number of retransmissions needed to get one successful transmission of a packet from source to destination node. We have

$$N_r = \frac{P_{fail}}{1 - P_{fail}}.$$

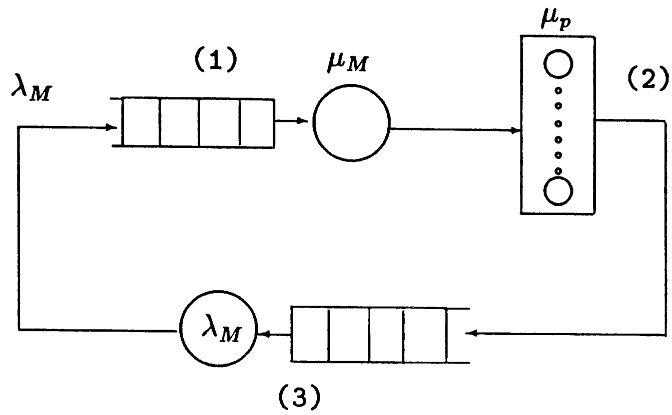
Define $T_{processing}$ to be the processing time in the link layer and $T_{timeout}$ to be the timeout period. We assume that the timeout period starts after the packet is transmitted from node 1(source node). Thus, the average end-to-end packet delay, $EEPD_{nn}$, is calculated as :

$$EEPD_{nn} = N_r \cdot (T_{processing} + T_1 + T_{timeout}) + \sum_{i=1}^{M} T_i. \tag{12}$$

3.2 Blocking Probability

3.2.1 Computation of P_{B_M}

If the number of packets in the destination node is equal to the maximum buffer size, the packet which is ready to be transmitted in node M will be blocked. To find P_{B_M}, we can model stage M as a closed queuing network shown in Fig(4).



Fig(4)

Let's denote the queue with service rate λ_M in Fig(4) to be queue 3. Let queue 1 and queue 2 be the transmission queue and the infinite server queue respectively. We define n_1,n_2,and n_3 to be the corresponding number of customers in queue 1, queue 2, and queue 3. Since this closed queueing network has a fixed and finite number of customers, K, circulating in it and each node consists of server(s) providing exponential service rate, we get the product form solution of the equilibrium distribution of customers in the closed system. From Gordon and Newell [9], we have

$$P(n_1, n_2, n_3) = \frac{1}{G(K)}\rho_1(n_1)\rho_2(n_2)\rho_3(n_3), \qquad n_1 + n_2 + n_3 = K. \tag{13}$$

where we have selected $\rho_3(n_3) = 1$ and as a consequence of this choice we get

$$\begin{aligned} \rho_1(n_1) &= \left(\frac{\lambda_M}{\mu_M}\right)^{n_1} \\ \rho_2(n_2) &= \left(\frac{\lambda_M}{\mu_M}\right)^{n_2} \cdot \frac{1}{n_2!} \end{aligned}$$

The normalization constant G(K) can be computed as:

$$\begin{aligned} G(K) &= \sum_{n_1+n_2+n_3=K} \rho_1(n_1)\rho_2(n_2)\rho_3(n_3) \\ &= \sum_{n=0}^{K}\sum_{m=0}^{K-n}\left(\frac{\lambda_M}{\mu_M}\right)^m \left(\frac{\lambda_M}{\mu_p}\right)^n \cdot \frac{1}{n!} \\ &= \sum_{n=0}^{K} \frac{1-\left(\frac{\lambda_M}{\mu_M}\right)^{K-n+1}}{1-\frac{\lambda_M}{\mu_M}} \cdot \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!} \end{aligned} \tag{14}$$

Thus, the blocking probability, P_{B_M}, is given as

$$P_{B_M} = \sum_{n=0}^{K} \frac{1}{G(K)}\left(\frac{\lambda_M}{\mu_M}\right)^n \left(\frac{\lambda_M}{\mu_p}\right)^{K-n} \frac{1}{(K-n)!} \tag{15}$$

In order to reduce the computational complexity , we need to find a recursive formula for computing P_{B_M}. From the equation above, we rewrite P_{B_M} as

$$\begin{aligned} P_{B_M} &= \frac{\sum_{n=0}^{K}\left(\frac{\lambda_M}{\mu_M}\right)^n \left(\frac{\lambda_M}{\mu_p}\right)^{K-n} \frac{1}{(K-n)!}}{\sum_{n=0}^{K} \frac{1-\left(\frac{\lambda_M}{\mu_M}\right)^{K-n+1}}{1-\frac{\lambda_M}{\mu_M}} \cdot \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}} \\ &= \frac{1-\frac{\lambda_M}{\mu_M}}{\frac{\sum_{n=0}^{K}\left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}}{\sum_{n=0}^{K}\left(\frac{\lambda_M}{\mu_M}\right)^{K-n} \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}} - \frac{\lambda_M}{\mu_M}} \end{aligned} \tag{16}$$

Let's introduce $f(K)$ as

$$f(K) = \frac{\sum_{n=0}^{K}\left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}}{\sum_{n=0}^{K}\left(\frac{\lambda_M}{\mu_M}\right)^{K-n} \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}}$$

$$= \frac{\sum_{n=0}^{K-1} \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!} + \left(\frac{\lambda_M}{\mu_p}\right)^K \frac{1}{K!}}{\frac{\lambda_M}{\mu_M} \sum_{n=0}^{K-1} \left(\frac{\lambda_M}{\mu_M}\right)^{K-n-1} \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!} + \left(\frac{\lambda_M}{\mu_p}\right)^K \frac{1}{K!}}$$

$$= \frac{f(K-1) + \left(\frac{\lambda_M}{\mu_p}\frac{1}{K}\right) \cdot g(K-1)}{\frac{\lambda_M}{\mu_M} + \left(\frac{\lambda_M}{\mu_p}\frac{1}{K}\right) \cdot g(K-1)} \tag{17}$$

where the $g(K-1)$ function is defined as

$$g(K-1) = \frac{\left(\frac{\lambda_M}{\mu_p}\right)^{K-1} \frac{1}{(K-1)!}}{\sum_{n=0}^{K-1} \left(\frac{\lambda_M}{\mu_M}\right)^{K-n-1} \left(\frac{\lambda_M}{\mu_p}\right)^n \frac{1}{n!}}$$

and

$$g(K-1) = \frac{\left(\frac{\lambda_M}{\mu_p}\frac{1}{K-1}\right) g(K-2)}{\frac{\lambda_M}{\mu_M} + \left(\frac{\lambda_M}{\mu_p}\frac{1}{K-1}\right) g(K-2)} \tag{18}$$

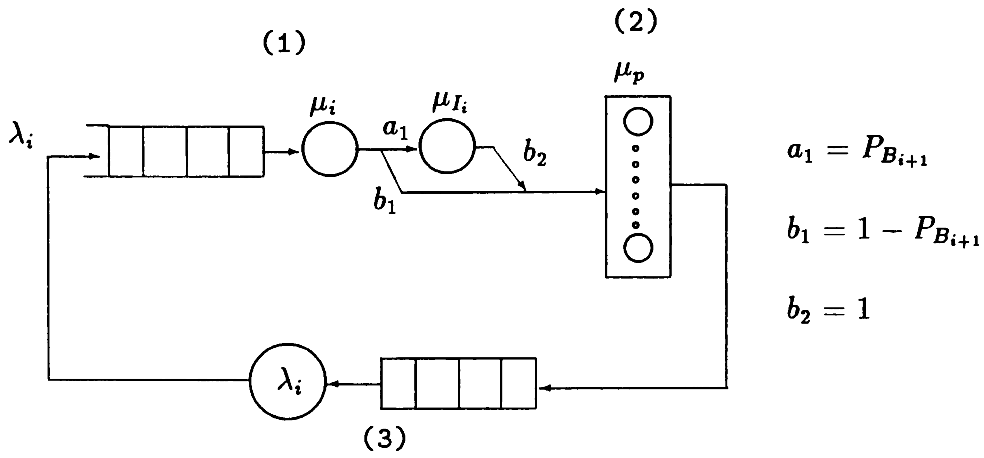
Thus, we find that

$$P_{B_M} = \frac{1 - \frac{\lambda_M}{\mu_M}}{f(K) - \frac{\lambda_M}{\mu_M}} \tag{19}$$

Now, we can easily compute P_{B_M} by using the recursive formula starting with the initial conditions : $f(0) = 1$ and $g(0) = 1$.

3.2.2 Approximate Solution of the Blocking Probability, P_{B_i}

If the number of packets held in node i+1 is equal to the maximum buffer size, the packet which is ready to be transmitted in node i will be blocked. To find P_{B_i} ,$i = 2, 3, \cdots, M-1$, we can model stage i as a closed queueing network shown in Fig(5).



Fig(5)

Let us denote the queue with service rate λ_i in Fig(5) to be queue 3. Let the transmission queue and the infinite server queue be queue 1 and queue 2 respectively. Since queue 1 has a Coxian 2 server, this network is not a product form closed queueing network. To find P_{B_i}, we can use Marie's method [5,6] to get an approximate solution of the blocking probability.

Let us denote N-(i) to be the aggregate network which is obtained by shorting queue i. The aggregate network is analyzed with a view to determining its flow equivalent server. This is achieved by "exponentiating" the aggregate network, i.e. each node is replaced by a simpler node with an exponentially distributed service time with the same mean as the original. This "exponentiation" is necessary in order to capitalize on the product form solution. The Marie's algorithm can be executed as following :

iteration #1

step 1: short queue 1.

Let P_{ij} be the probability that a customer leaving station i enters station j. Let $\lambda_1(n)$

be the rate at which customers arrive at the shorting point if there are K-n jobs in the aggregate network or n jobs in queue 1. Then,

$$\begin{aligned}
\lambda_1(n) &= \sum_{i=2}^{3} P_{i1} \cdot \mu_i' \cdot P(n_i > 0) \qquad , \mu_2' = \mu_p \quad \textit{and} \quad \mu_3' = \lambda_i \\
&= \lambda_i \cdot P(n_3 > 0) \\
&= \lambda_i \cdot \frac{G_{N-1}(K-n-1)}{G_{N-1}(K-n)} \qquad (20) \\
n &= 0, 1, \cdots, K-1. \qquad \lambda_1(K) = 0.
\end{aligned}$$

where

$$\begin{aligned}
G_{N-1}(K-n-1) &= \sum_{n_2+n_3=K-n-1} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \cdot \frac{1}{n_2!} \\
&= \sum_{n_2=0}^{K-n-1} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \cdot \frac{1}{n_2!}
\end{aligned}$$

Thus, we get

$$\begin{aligned}
\lambda_1(n) &= \lambda_i \cdot \frac{\sum_{n_2=0}^{K-n-1} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \frac{1}{n_2}}{\sum_{n_2=0}^{K-n} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \frac{1}{n_2!}} \\
&= \lambda_i \left(1 - \frac{\left(\frac{\lambda_i}{\mu_p}\right)^{K-n} \frac{1}{(K-n)!}}{\sum_{n_2=0}^{K-n} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \frac{1}{n_2!}} \right) \\
&= \lambda_i \cdot [1 - h(K-n)]. \qquad (21)
\end{aligned}$$

where

$$
\begin{aligned}
h(K-n) &= \frac{\left(\frac{\lambda_i}{\mu_p}\right)^{K-n} \frac{1}{(K-n)!}}{\sum_{n_2=0}^{K-n} \left(\frac{\lambda_i}{\mu_p}\right)^{n_2} \frac{1}{n_2!}} \\
&= \frac{\frac{\lambda_i}{\mu_p}\frac{1}{K-n} h(K-n-1)}{1+\frac{\lambda_i}{\mu_p}\frac{1}{K-n} h(K-n-1)} \qquad (22) \\
h(0) &= 1.
\end{aligned}
$$

Now,node 1 is analyzed as an $\lambda(n)/C_2/1/K$ queue. From Marie's equations [7], we obtain $v_1^{(1)}(n), n = 1, 2, \cdots, K$, the conditional throughput of node 1 in the first iteration :

$$
v_1^{(1)}(1) = \frac{\lambda_1(1)\cdot\mu_i\cdot(1-P_{B_{i+1}}) + \mu_i\cdot\mu_{I_i}}{\lambda_1(1) + P_{B_{i+1}}\cdot\mu_i + \mu_{I_i}} \qquad (23)
$$

$$
v_1^{(1)}(n) = \frac{\lambda_1(n)\cdot\mu_i\cdot(1-P_{B_{i+1}}) + \mu_i\cdot\mu_{I_i}}{(\lambda_1(n) + \mu_i + \mu_{I_i}) - v_1^{(1)}(n-1)} \qquad ,n > 1. \qquad (24)
$$

step 2: short queue 2.

Let us exponentiate nodes 1 and 3, and denote N-(2) to be the aggregate system . From Gordon and Newell [9] , we get

$$
P(n_1, n_3) = \frac{1}{G_{N-2}(K-n)} \rho_1(n_1)\rho_3(n_3) \qquad (25)
$$

where we select $\rho_3(n_3)$ to be 1 and

$$
\rho_1(n_1) = \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)}
$$

The normalization constant is calculated as:

$$
G_{N-2}(K-n) = \sum_{n_1+n_3=K-n} \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)}
$$

$$= \sum_{n_1=0}^{K-n} \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)} \tag{26}$$

Thus, the arrival rate at the shorting point is

$$\begin{aligned} \lambda_2(n) &= \sum_{n_1=1}^{K-n} P_{12} \cdot v_1^{(1)}(n_1) \cdot P(n_1, n_3) \\ &= \sum_{n_1=1}^{K-n} v_1^{(1)}(n_1) \cdot \frac{1}{G_{N-2}(K-n)} \cdot \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)} \qquad (27) \\ for \quad n &= 0, 1, \cdots, K-1. \qquad \lambda_2(K) = 0. \end{aligned}$$

Since queue 2 is an infinite server queue with exponential service rate, we have the conditional throughput of node 2

$$v_2^{(1)}(n) = n \cdot \mu_p \qquad , n = 1, 2, \cdots, K. \tag{28}$$

step 3: short queue 3.

Let N-(3) be the aggregate system . From Gordon and Newell [9] , we get

$$P(n_1, n_2) = \frac{1}{G_{N-3}(K-n)} \rho_1(n_1) \rho_2(n_2) \tag{29}$$

where

$$\rho_1(n_1) = \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)} \qquad , \rho_2(n_2) = \prod_{j=1}^{n_2} \frac{\lambda_i}{v_2^{(1)}(j)}$$

The normalization constant is :

$$\begin{aligned} G_{N-3}(K-n) &= \sum_{n_1+n_2=K-n} \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)} \prod_{l=1}^{n_2} \frac{\lambda_i}{v_2^{(1)}(l)} \\ &= \sum_{n_1=0}^{K-n} \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1^{(1)}(j)} \prod_{l=1}^{K-n-n_1} \frac{\lambda_i}{v_2^{(1)}(l)} \qquad (30) \end{aligned}$$

where

$$\prod_{j=1}^{0} \frac{\lambda_i}{v_1^{(1)}(j)} = 1, \qquad \prod_{l=1}^{0} \frac{\lambda_i}{v_2^{(1)}(l)} = 1.$$

Thus, the arrival rate at the shorting point is :

$$\begin{aligned} \lambda_3(n) &= \sum_{n_2=1}^{K-n} P_{23} \cdot v_2^{(1)}(n_2) \cdot P(n_1, n_2) \\ &= \sum_{n_2=1}^{K-n} v_2^{(1)}(n_2) \cdot \frac{1}{G_{N-3}(K-n)} \prod_{j=1}^{K-n-n_2} \frac{\lambda_i}{v_1^{(1)}(j)} \prod_{l=1}^{n_2} \frac{\lambda_i}{v_2^{(1)}(l)} \qquad (31) \\ for \quad n &= 0, 1, \cdots, K-1. \qquad \lambda_3(K) = 0 \end{aligned}$$

Since queue 3 is a single server queue with exponential service rate, we have the conditional throughput

$$v_3^{(1)}(n) = \lambda_i \qquad , n = 1, 2, \cdots, K. \qquad (32)$$

Having completed iteration 1, we have an approximate expression for the conditional throughput of node i, $v_i^{(1)}(n_i)$, for $n_i = 1, 2, \cdots, K$. Let $P_i(n)$ be the equilibrium state probabilities that there are n_i customers in node i. For a $\lambda_1(n)/C_2/1/K$ queue considered in equilibrium behavior [7], we can compute $P_1(n_1)$ as following :

$$v_1(n_1) \cdot P_1(n_1) = \lambda_1(n_1 - 1) \cdot P_1(n_1 - 1) \qquad (33)$$

$$v_1(1) \cdot P_1(1) = \lambda_1(0) \cdot P_1(0) \qquad n_1 = 1, 2, \cdots, K. \qquad (34)$$

and

$$\sum_{n_1=0}^{K} P_1(n_1) = 1.$$

For the infinite server queue, we can compute $P_2(n_2)$ as following :

$$\lambda_2(n_2) \cdot P_2(n_2) = v_2(n_2 + 1) \cdot P_2(n_2 + 1) \qquad (35)$$

$$P_2(n_2) = \prod_{j=0}^{n_2-1} \frac{\lambda_2(j)}{v_2(j+1)} \cdot P_2(0) \qquad , v_2(j+1) = (j+1) \cdot \mu_p \tag{36}$$

and

$$\sum_{n_2=0}^{K} P_2(n_2) = 1.$$

For queue 3, we can compute $P_3(n_3)$ as following :

$$\lambda_3(n_3) \cdot P_3(n_3) = v_3(n_3+1) \cdot P_3(n_3+1) \tag{37}$$

$$P_3(n_3) = \prod_{j=0}^{n_3-1} \frac{\lambda_3(j)}{v_3(j+1)} \cdot P_3(0) \qquad , v_3(j+1) = \lambda_i \tag{38}$$

and

$$\sum_{n_3=0}^{K} P_3(n_3) = 1.$$

The iteration steps are repeated until the following termination conditions are satisfied.

$$\left| \frac{K - \sum_{i=1}^{3} \overline{n_i}}{K} \right| < \varepsilon \tag{39}$$

$$\left| \frac{\frac{t_i}{x_i} - \frac{1}{3} \sum_{j=1}^{3} \frac{t_j}{x_j}}{\frac{1}{3} \sum_{j=1}^{3} \frac{t_j}{x_j}} \right| < \varepsilon \qquad , i = 1, 2, 3. \tag{40}$$

where

$$\overline{n_i} = \sum_{n_i=1}^{K} n_i \cdot P_i(n_i)$$

$$t_i = \sum_{n_i}^{K} \lambda_i(n_i) \cdot P_i(n_i)$$

For the network shown in Fig(5), there is only one node with nonexponential service time distribution. In this case, we just need one iteration to satisfy the termination conditions. Now, we get the conditional throughput $v_i(n_i), i = 1, 2, 3$. Thus, the equilibrium probability distribution $P(n_1, n_2, n_3)$ is computed as :

$$P(n_1, n_2, n_3) = \frac{1}{G(K)} \cdot \rho_1(n_1) \cdot \rho_2(n_2) \cdot \rho_3(n_3) \tag{41}$$

$$\begin{aligned} \rho_1(n_1) &= \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1(j)} \\ \rho_2(n_2) &= \prod_{j=1}^{n_2} \frac{\lambda_i}{v_2(j)} \\ \rho_3(n_3) &= \prod_{j=1}^{n_3} \frac{\lambda_i}{v_3(j)} \end{aligned} \tag{42}$$

The normalization constant is computed as :

$$G(K) = \sum_{n_1+n_2+n_3=K} \prod_{j=1}^{n_1} \frac{\lambda_i}{v_1(j)} \prod_{l=1}^{n_2} \frac{\lambda_i}{v_2(l)} \prod_{i=1}^{n_3} \frac{\lambda_i}{v_3(i)} \tag{43}$$

Thus, the approximate solution of the blocking probability ,$P_{B_i}, i = 2, 3, \cdots, M-1$, is calculated as :

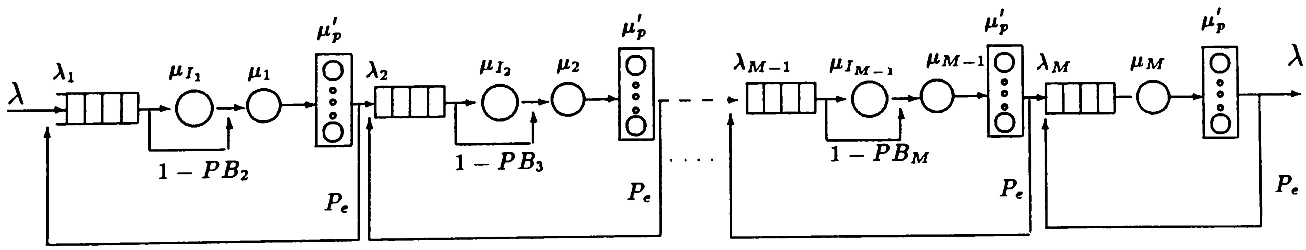
$$\begin{aligned} P_{B_i} &= \sum_{n_1+n_2=K} P(n_1, n_2, 0) \\ &= \sum_{n=0}^{K} P(n, K-n, 0) \\ &= \frac{\sum_{n=0}^{K} \prod_{j=1}^{n} \frac{\lambda_i}{v_1(j)} \cdot \left(\frac{\lambda_i}{\mu_p}\right)^{K-n} \cdot \frac{1}{(K-n)!}}{\sum_{n=0}^{K} \sum_{m=0}^{K-n} \prod_{j=1}^{m} \frac{\lambda_i}{v_1(j)} \cdot \left(\frac{\lambda_i}{\mu_p}\right)^{n} \cdot \frac{1}{n!}} \end{aligned} \tag{44}$$

4 Link-by-link Error Control

In the link-by-link error control approach, error recovery is done by two adjacent nodes along the virtual circuit. Each node of the virtual circuit will buffer a packet which has been transmitted until an ACK message for the packet is received from the next node. If the buffer of the next node is full, the packet which is ready to transmit in the node will be blocked and the link will sit idle until a receive ready signal from the next node is received. If a node receives a correct packet, it will send an ACK message to the preceding node. If an ACK message of a transmitted packet is not received within a timeout period, the packet will be retransmitted. We assume that a timeout occurs if the packet is in error.

4.1 Link level model

A link level virtual circuit model of the link-by-link error recovery scheme is shown in Fig(6).



Fig(6) Link level model of the link-by-link error recovery scheme

Assume that the effective arrival rate of each node is a Poisson process with rate $\lambda_i, i = 1, 2, \cdots, M$. We also assume that the idle time due to a packet being blocked is exponentially distributed with mean $1/\mu_{I_i}, i = 1, 2, \cdots, M$ and $1/\mu_{I_M}$ is equal to 0. The packet transmission time at each node is assumed exponentially distributed with mean $1/\mu_i, i = 1, 2, \cdots, M$. Since the transmitted packets have to sit waiting for an ACK message from the next node, the customers in the infinite server queue represent the packets for which a timeout period has been started. The time $1/\mu'_p$ is equal to the timeout period which is calculated as :

$$T_{timeout} = \frac{1}{\mu'_p} = 2 \cdot \frac{1}{\mu_p} + T_{processing} + T_{NAK} \tag{45}$$

where $1/\mu_p$ is the propagation delay time through a link , $T_{processing}$ is the processing time of the link layer , T_{NAK} is the NAK message transmission time. Let K be the maximum buffer size of a node and $P_{B_i}, i = 2, 3, \cdots, M$, be the probability that the buffer of node i is full. Since we assume that the first node has enough buffer to store all new packets, P_{B_1} is equal to 0. We denote P_e to be the probability that a packet may be in error when it passes through a link of the virtual circuit. In steady state, we get

$$\lambda_i = \frac{\lambda_{VC}}{1 - P_e} \qquad , i = 1, 2, \cdots, M. \tag{46}$$

Let N_r^i be the average number of retransmissions needed to successfully transmit a packet from node i to node i+1. Thus,

$$N_r^i = \frac{P_e}{1 - P_e} \qquad , i = 1, 2, \cdots, M. \tag{47}$$

Now introduce S_i to be the average time taken to get a successful transmission of a packet from node i to node i+1. Thus, we have

$$S_M \;=\; N_r^M \cdot \left(\frac{1}{\mu_M} + \frac{1}{\mu'_p} + T_{processing} \right) + \frac{1}{\mu_M} + T_{processing} + \frac{1}{\mu_p} \tag{48}$$

$$S_i = N_r^i \cdot \left(\frac{1}{\mu_i} + \frac{1}{\mu'_p} + P_{B_{i+1}} \cdot \frac{1}{\mu_{I_i}} + T_{processing}\right) +$$

$$\frac{1}{\mu_i} + \frac{1}{\mu_p} + \frac{1}{\mu_{I_i}} + T_{processing} \qquad ,i = 1, 2, \cdots, M-1. \tag{49}$$

$$\frac{1}{\mu_{I_i}} = S_{i+1} \qquad ,i = 1, 2, \cdots, M-1. \tag{50}$$

The average end-to-end packet delay of the link-by-link error recovery scheme, $EEPD_{ll}$, can be computed as:

$$EEPD_{ll} = \sum_{i=1}^{M} S_i. \tag{51}$$

4.2 Blocking Probability

The total number of packets buffered at a node is the sum of the number of packets in the transmission queue and the number of packets in the infinite queue which represent the packets that have started a timeout period. To find P_{B_M}, we can use the model shown in Fig(4) by changing μ_p to μ'_p. The complete methods have been shown in section 3.2.1.

To find $P_{B_i}, i = 2, 3, \cdots, M-1$, we can use the model shown in Fig(5). We need to modify μ_p to μ'_p and $1/\mu_{I_i}$ to S_{i+1}. The approximate solution of the blocking probability ,P_{B_i}, of the link-by-link error recovery scheme can be calculated by following the methods shown in section 3.2.2.

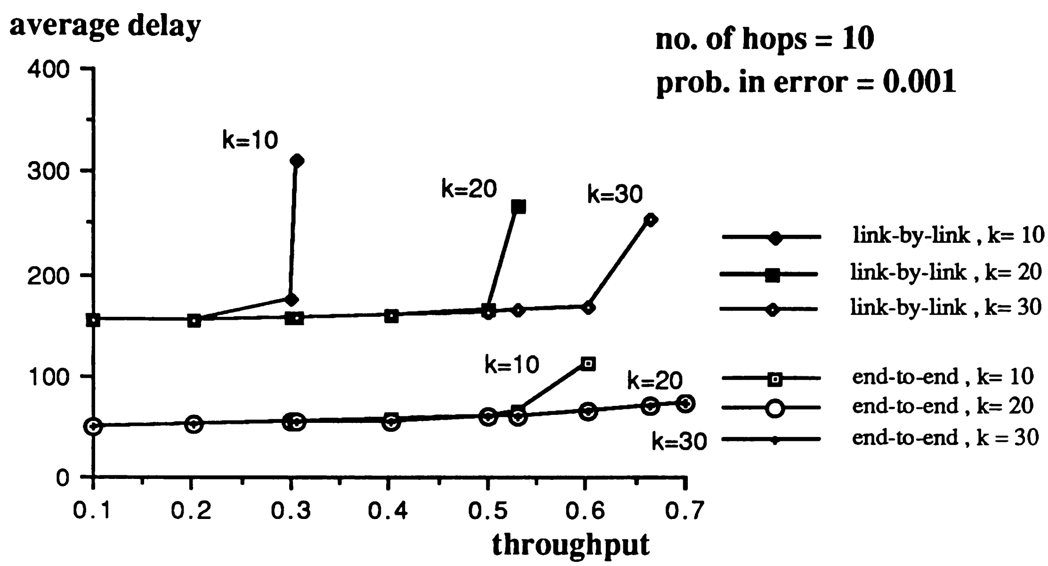
5 Results and Conclusion

For the numerical results presented here, the average packet length is set to 10,000 bits, the link capacity is 150 M bps, the link processing time is 0.7 msec and the propagation delay is 0.25 msec/link. For the figures (7),(8),(9), and (10), a unit scale in the vertical axis represents a packet transmission time.

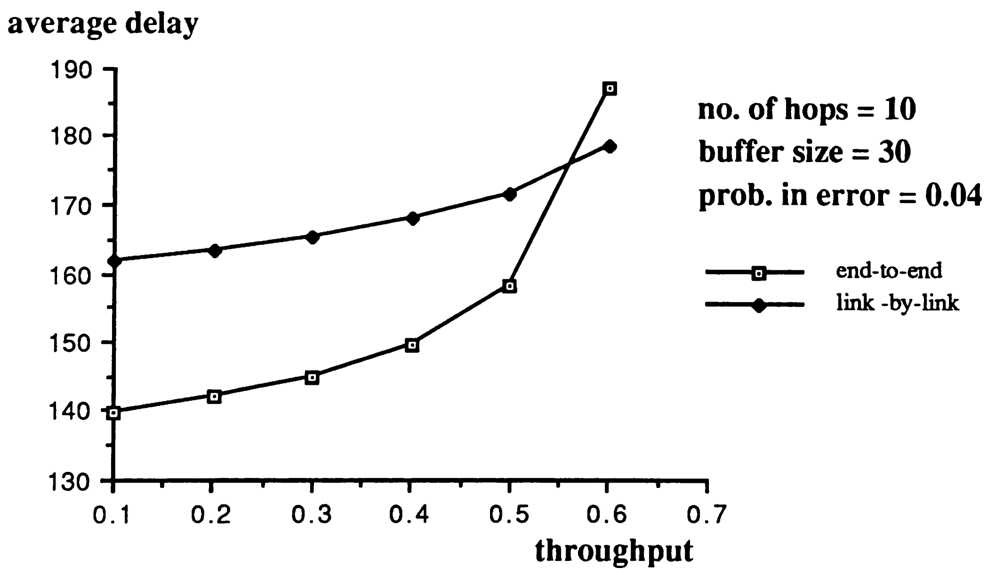
In Fig(7), we compare the average packet delay along the virtual circuit for the end-to-end and the link-by-link error recovery schemes and find the effect of the buffer size on the average packet delay. The results indicate that the end-to-end error recovery scheme has shorter delay if the packet error rate is low. This is due to the processing overhead in the link layer of the link-by-link error recovery scheme. The reason why, in Fig(7), the link-by-link approach will saturate very quickly is due to the finite buffer size problem. Since the buffer size(K=10) is small, the blocking probability will become an important factor to the packet delay. If we increase the packet error rate, we find, in Fig(8), that under light traffic the end-to-end approach has shorter delay but under heavy traffic the link-by-link approach can have better performance.

In Fig(9), we fix the number of hops, the buffer size, and the throughput to see the effect of the packet error rate to the average packet delay of the two error recovery schemes. We find that if the packet error rate is small enough, the end-to-end error recovery scheme can have shorter delay. Fig(10) shows us the effect of the number of hops of the virtual circuit to the average packet delay of the two error recovery schemes. The result shows that under a paticular condition, with an adequate number of hops of the virtual circuit , the end-to-end error recovery scheme can have shorter packet delay.

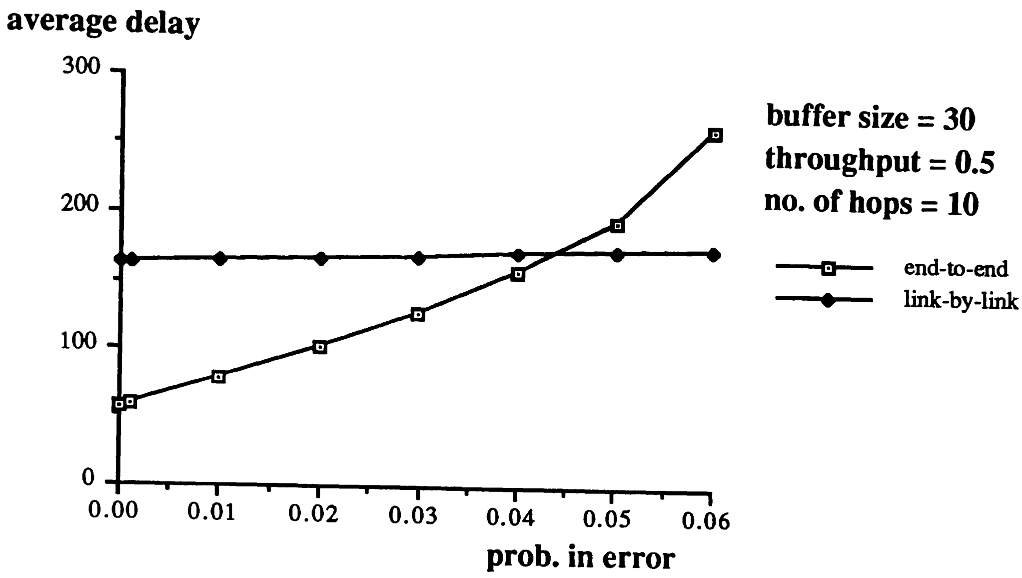
From the above analysis, we find that the average end-to-end packet delay is affected by the packet error rate, the buffer size,the number of hops along the virtual circuit,the traffic along the virtual circuit, and the processing overhead in the link layer. Thus, in a high speed, low error rate, optical fiber network, the end-to-end error recovery scheme can achieve shorter average end-to-end packet delay.



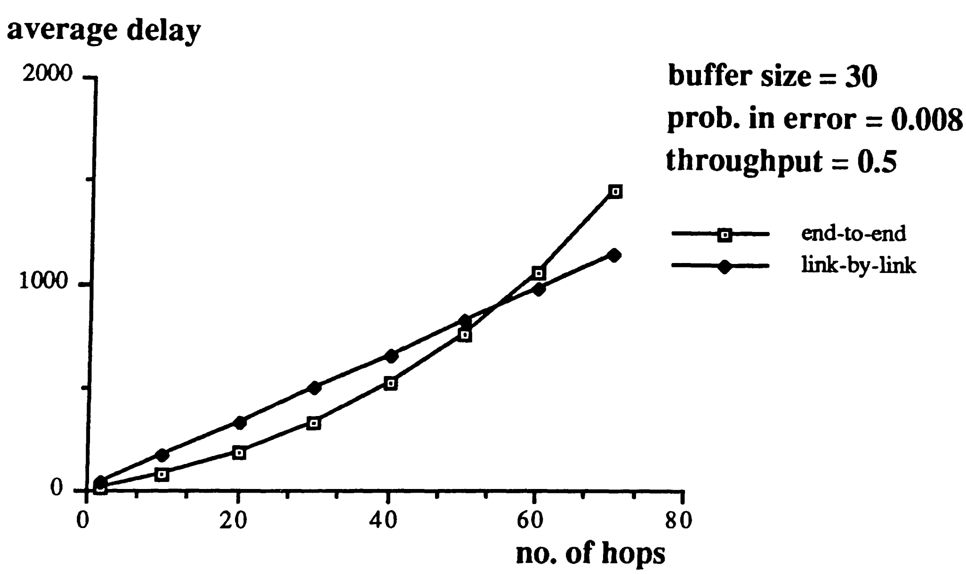
Fig(7) Average end-to-end packet delay with low packet error rate



Fig(8) Average end-to-end packet delay with high packet error rate



Fig(9) Effect of the packet error rate



Fig(10) Effect of the number of hops

References

[1] Amit Bhargava, James F. Kurose, Don Towsley, Guy Van Leemput "*Performance Comparison of Error Control Schemes in High Speed Computer Communication Networks*" , IEEE Journal on Selected Areas in Communications, Dec.,1988.

[2] Tatsuya Suda, Naoya Watanabe "*Evaluation of Error Recovery Schemes for a High-Speed Packet Switched Network : Link-by-Link Versus Edge-to-Edge Schemes*" ,1988,IEEE.

[3] Marek I. Irland and Guy Pujolle "*Comparison of Two Packet-Retransmission Techniques*" ,IEEE Transactions on Information Theory, Jan. 1980.

[4] Michael C. Pennotti and Mischa Schwartz "*Congestion Control in Store and Forward Tandem Links*" ,IEEE Transactions on Communications, Dec. 1975.

[5] Raymond A. Marie "*An Iterative Method for General Queuing Networks*" , I.R.I.S.A. INSA 35031 RENNES FRANCE.

[6] Raymond A. Marie "*An Approximate Analytical Method for General Queuing Networks*" , I.R.I.S.A.- I.N.S.A. B.P. 14A 35031 RENNES FRANCE.

[7] Raymond A. Marie "*Calculating Equilibrium Probabilities For* $\lambda(n)/C_k/1/N$ *Queues* " , ACM,1980.

[8] Leonard Kleinrock "*Queneing Systems Volume I*", John Wiley, 1976.

[9] W.J. Gordon and G.F. Newell, "*Closed Queueing Systems with Exponential Servers* ", Operations Research, 15, 1967.