An analysis of the impact of suspending cell discarding in TCP-over-ATM

Shunsaku NAGATA, Naotaka MORITA, Hiromi NOGUCHI, and Kou MIYAKE
NTT Network Service Systems Labs
3-9-11 Midori-Cho Musashino-City, Tokyo 180-8585, Japan
Tel: +81-422-59-3404 Fax: +81-422-59-3494
e-mail Nagata.Shunsaku@nslab.ntt.co.jp

Abstract

A cell-discard method has been developed for use in networks supporting TCP-over-ATM traffic with an unspecified bit rate. In this "concentrated cell discard" (CCD) method, when an ATM switch becomes congested, consecutive cells are discarded from the front of the buffer, and cell discarding is suspended for a specified period of time. This method prevents TCP timeouts, which are the main reason for the decreasing TCP throughput in today's TCP-over-ATM networks, and shortens the duration of the congestion by avoiding the long delay experienced with the TCP fast-retransmission algorithm before the sender's TCP receives duplicate acknowledgments. Even though CCD does not consider TCP packet size or boundaries, the throughput with CCD was found by computer simulation to actually be higher than with early packet discard.

Key words: ATM, TCP Reno, cell discard, early packet discard, UBR

1. Introduction

To support the growing number of Internet users and to manage the wider network bandwidth required by faster applications, researchers are developing many methods to support the transmission control protocol (TCP), which is used for Internet traffic, over asynchronous-transfer-mode (ATM) networks. Although TCP already has its own end-to-end flow control using only TCP timeout retransmission, it was improved by adding fast retransmission and fast recovery (TCP Reno [1]). An attractive way of applying ATM to the underlying TCP layer is to use unspecified-bit-rate (UBR) transmission, even though this requires a larger buffer. With UBR, the network can achieve a high throughput in the TCP layer as long as the ATM switch discards cells in an efficient manner. However, in UBR, when there is no space in the buffer, cell discard is usually done on the input side of the buffer. This so-called tail discarding (TLD) is commonly used because it is simple to implement. However, when it is implemented in an ATM switch in a TCP-over-ATM network, two problems arise.

Problem 1: Continuous cell discarding

ATM switches that use TLD discard cells only when the queue length is equal to or greater than the buffer size or threshold. Therefore, when the input rate to the buffer is larger than the output rate, the time between cell discards is very short, and many of the TCP packets that were sent within the same TCP window are dropped. As a result, the probability of a TCP timeout occurring is very high (as will be explained in Sec. 2). In general, where retransmission of dropped TCP packets is done only by TCP fast-retransmission and recovery, the throughput is higher than when it is done by TCP timeout. This is because the time taken for TCP to recognize network congestion in TCP timeout is much longer than in TCP fast-retransmission and recovery, and because the cwnd (congestion window) just after retransmission of a dropped packet in TCP timeout is smaller than in TCP fast-retransmission and recovery. Consequently, TCP throughput is reduced in this case.

Problem 2: Long time wasted passing through buffer

If one TCP packet is discarded at the tail-end of the buffer, it will take a period of time corresponding to the buffer size for the following few consecutive packets to pass through the buffer. This delays the start of the fast-retransmission fast-recovery algorithm at the sender, because the sender's TCP recognizes packet discard only when it receives the acknowledgments (ACKs) to the packets following the dropped packet. This reduces the throughput in the TCP layer because the queue length continues increasing and many packets are discarded during the delay period. In TCP over UBR, an ATM switch should have a buffer large enough to prevent overflow; consequently, this waiting delay is very long. Many of today's ATM switches have a buffer capacity of more than 100,000 cells. The corresponding time wasted passing through the buffer is more than 270 ms, which is larger than the minimum size of the TCP retransmission timeout (RTO) [2], when the output rate of the buffer is about 155 Mbps. Consequently, the probability of a TCP timeout occurring is very high.

Many ATM switches now use early packet discard (EPD) [3]. EPD discards all the useless cells from the input side of the buffer, i.e., those composing TCP packets that will be discarded in the receiver's TCP. As a result, the throughputs in the ATM layer are very good. However, this approach is not optimal for discarding cells in a TCP-over-ATM network because it does not solve either of the above problems. As a result, EPD does not have any measure for avoiding TCP timeout. Another approach is the "drop from front (DFF)" method [4], which discards cells from the front of the buffer when the buffer is full. This method solves problem 2 but not problem 1.

We have thus developed a new approach to discarding cells. When an ATM switch becomes congested, it initially discards a specified number of cells. To prevent a TCP timeout from occurring, cell discarding is then suspended for a specified time, even though the ATM switch becomes congested again. In addition, to prevent buffer overflow during the time when cell discarding is suspended, cells are discarded from the front of the buffer. This keeps the period of congestion short because the packets following the discarded packets are output quickly. We call this method "concentrated cell discard (CCD)".

This paper describes our proposed CCD method and discusses its performance. In section 2 we describe the relationship between TCP throughput and the number of discarded packets sent within the same congestion window (cwnd). In section3, we describe our proposed CCD in detail, where we describe the procedure of CCD, the method of calculating parameters, the method of setting these parameters in a real ATM switch, and the robustness of CCD. In section 4, we make perform some simulations and evaluate the effectiveness of CCD.

2. Relationship between number of discarded packets and TCP throughput

The most basic way of retransmitting a discarded TCP packet is for the sender to retransmit it after the TCP retransmission timer monitoring the Round Trip Time (RTT) has expired. This method, called TCP timeout, is the only retransmission method available in TCP Tahoe [1], which is the basis of TCP flow control. Because the size of the TCP cwnd is decreased by the size of one TCP packet during congestion and the timer is usually set significantly longer than RTT to avoid unnecessary retransmission, throughput is seriously degraded if a packet is discarded.

TCP Reno was developed to achieve higher throughput than is possible with TCP Tahoe. It features fast retransmission and fast recovery. A lost packet is retransmitted when TCP Reno deduces from a small number (normally three) of consecutive duplicate acknowledgments (ACKs) that a packet has been lost. However, when fast retransmission and fast recovery do not work, a dropped packet is transmitted by TCP timeout, greatly decreasing throughput. Therefore, it is important in TCP Reno that TCP timeouts do not occur. Next, we review the conditions for preventing TCP timeout; these have already been described in detail [1]. If only one TCP packet is discarded, TCP can receive three duplicate ACKs for dropped packets and TCP timeout does not occur only when Eq. (1) holds.

$$Wcwnd >= 4 \text{ (packets)}, \quad (1)$$

where Wcwnd is the size of the congestion window in the sender's TCP “immediately before” fast retransmission was initiated.

If two TCP packets are discarded, TCP can receive three duplicate ACKs for dropped packets and TCP timeout does not occur only when Eq. (2) holds.

$$Wcwnd >= 10 \text{ (packets)} \quad (2)$$

If three TCP packets are discarded, TCP can receive three duplicate ACKs for dropped packets and TCP timeout does not occur only when Eqs. (3) and (4) hold.

$$Wcwnd >= 20 \text{ (packets)} \quad (3)$$

$$dp3 >= 2+(Wcwnd \sim 3/4) \text{ (packets)}, \quad (4)$$

where dp3 is the interval between the first and second discarded packets.

Therefore, if an ATM switch can control the size of the congestion window, it can ensure that a TCP timeout does not occur. However, this is impossible with today's technology. Our CCD ensures that a TCP timeout is not caused by suspending cell discarding for a specified period of time, which must be longer than the time taken to output ten TCP packets, as long as buffer overflow does not occur. The time must also be longer than it takes to output TCP packets whose total size is half the maximum size of the congestion window (65,536 bytes). Consequently, two-packet discarding cannot occur when Wcwnd is less than ten packets, and discards of three or more packets can never occur.

3. Proposed CCD method

3.1 Overview

In CCD, when a cell arrives at a congested buffer containing Tq cells,

- D consecutive cells are discarded from the output side of the buffer without any consideration of the TCP packet boundaries.
- Cell discard is suspended for a time period Tt, during which the switch outputs all the packets in a TCP window if the cells of two consecutive TCP packets from that window have been discarded.

This concentrated cell discard (CCD) is schematically illustrated in Fig. 1.

This method solves problem 1 by suspending cell discarding for time Tt and it solves problem 2 by discarding cells from the front of the buffer.

3.2 Calculation of parameters

In this section, we discuss the appropriate ranges for the parameters (Tq, D, Tt,) defined above for our method to operate effectively. To obtain high throughput in the TCP layer, we must prevent TCP timeout, as explained in Sec. 2. The following equalities and inequalities are the conditions necessary for preventing TCP timeout.

(1) Number of cells discarded in CCD: D

First, to avoid three consecutive packets being discarded, which causes a TCP timeout, D should satisfy Eq. (5), in which ¿ should be within the range expressed by Eq. (6); Nvc is the number of VC connections using

the congested buffer; and Pcell is the number of cells in one TCP packet.

D <= ¿ ~Pcell ~Nvc (cells) (5)
(1/Pcell) <= ¿ <1 (6)

Note that Eq. (5) requires that the number of consecutively discarded cells be fewer than the number of cells in one packet because the packet boundary is not taken into account. If ¿is greater than (1/Pcell), the probability of discarding two consecutive TCP packets is high. When two consecutive TCP packets are discarded, a TCP timeout does not occur if Wcwnd is larger than ten. If ¿is greater than 1, there is the probability of discarding three consecutive TCP packets, as is the probability of a TCP timeout occurring. Conversely, if ¿ is less than 1/Ptcp, there must be connections in which no ATM cells are discarded and in which the congestion window does not decrease during congestion, which means that the queue of the ATM switch must be very long.

(2) Time when the buffer suspends cell discarding: Tt

To avoid a TCP occurring, we set Tt as follows. When one TCP packet is discarded, we can say from Eq. (1) that a TCP timeout must occur if the size of the congestion window is smaller than four packets. But it is difficult to control the size of the TCP congestion window just after the occurrence of a cell discard caused by the ATM layer. Therefore, a method of eliminating the possibility of TCP timeout caused by the discarding of one TCP packet is being studied.

When two consecutive TCP packets are discarded, a TCP timeout does not occur if Wcwnd is larger than ten packets (from Eq. (2)). To avoid causing a TCP timeout, Tt should satisfy Eq. (7), while Pcell should be the number of cells in a TCP packet.

Tt >= Nvc ~9 ~Pcell (cells) (7)

When a congested ATM switch satisfies this equation, two consecutive TCP timeout does not occur if they are sent from a connection whose cwnd is less than ten packets.

When three consecutive TCP packets are discarded, we can say that a TCP timeout will occur if Wcwnd and dp3 do not satisfy Eqs. (3) and (4), respectively. To avoid this situation in CCD, Tt is set large enough that three-packet discarding does not occur for packets sent from servers with the same size cwnd, as represented in Eq. (8).

Tt >= Nvc ~(65536)/53 ~(1/2) (cells) (8)

We assume that the maximum size of cwnd is 65,536 (bytes) in Eq. (8). In addition, to avoid buffer overflow, Tt should be the smallest value that satisfies Eqs. (7) and (8). Therefore, the best value of Tt is

Tt = min[Nvc ~(65536)/53 ~(1/2), Nvc ~9 ~Pcell] (cells). (9)

(3) Threshold of CCD: Tq

With CCD, buffer overflow may occur because once CCD occurs, it suspends cell discarding for a time period Tt even though the queue length exceeds the buffer size. Therefore, CCD must inform TCP of congestion in an ATM switch as soon as possible. It does this by discarding ATM cells from the front of the congested buffer, so there is no waiting time in the buffer before TCP deduces that a packet has been lost from the consecutive duplicate ACKs sent in response to the packets following the discarded packet. However, this does not eliminate the possibility of a buffer overflow. It may be possible to prevent buffer overflow by decreasing the number of cells in the buffer (parameter Tq). However, it is difficult to regulate Tq because Tq cannot easily be calculated. In addition, the queue length is affected by RTT, the input rate, and the output rate from equalities (10) to (11). When installing CCD in an ATM switch, it is difficult to get correct values because these parameters vary over time. Moreover, the method of regulating an adequate value of Tq is still under study.

A method that guarantees to prevent buffer overflow is now being investigated. Consequently, we can say that, if CCD is used, Eqs. (5) to (8) hold, so buffer overflow does not occur and TCP timeout does not occur.

3.3. Procedure and effectiveness of CCD

First, we analyze the procedure and effectiveness of CCD and EPD with a generic network model (Fig. 2) and a symmetrical congestion model in sections (a) and (b).

The parameters we used to simulate congestion were as follows.

– Ptcp: TCP packet length (= 1500 bytes)
– Pcell: number of cells in one TCP packet (= 30 cells)
– B: buffer length (= 2000 cells)
– Nvc: number of connections sharing a buffer (= 10)
– W(t): total size of server cwnd immediately before a particular time t (packets)
– Q(t): queue length of congested ATM switch immediately before a particular time t (packets)
– D: number of consecutive cells discarded (cells)
– Tq: threshold of queue length for which EPD or CCD occurs (cells)
– Tt: threshold of time between cell discards where CCD occurs (cells)
– input rate: input rate to congested buffer for one connection (=30 Mbps)
– output rate: output rate from congested buffer for one connection (=15 Mbps)
– RTT (=5 ms)

To simplify the discussion in this section, we make the following assumptions.

(1) The network model is a client-server model, and all the

clients start simultaneously. Consequently, cell interleaving occurs in the buffer of the bottlenecked ATM switch (ATM-SW1 in Fig. 2), and the actions of all the servers and clients are the same.

(2) During congestion, the time taken for a server's TCP to output all the packets within the cwnd is larger than the RTT ((W(t)-1) ~Pcell ~53 ~8/input rate >= RTT).

Figures 3 and 4 show the action of server 1's TCP and the queue length of ATM-SW1 when CCD and EPD are used, respectively. In both cases, the queue grows longer than the threshold of Tq when ATM-SW1 receives TCP packet 4 (t1). At t1, the CCD buffer discards one ATM cell from TCP packet 1, from the output side of the buffer if the threshold of Tq is about 1000 (cells). At t1, the EPD buffer discards all the cells in TCP packet 4 from the input side of the buffer. Even though we presume that all the server actions and all the client actions are the same in this section, we focus on the connection between server 1 and client 1 in Figs. 3 and 4.

(a) ATM-SW1 uses CCD

As illustrated in Fig. 3, at t4, when one cell from TCP packet 4 arrives in the CCD buffer of ATM-SW1, the queue length is equal to Tq, so the buffer discards one cell from TCP packet 1. Next, the server's TCP receives 3 ACKs with the same sequence number (i.e., duplicate ACKs) that were sent in response to TCP packets 2 to 4 by client 1. The size of the cwnd of server 1 does not change. After receiving the third duplicate ACK, the server's TCP changes its state to fast retransmission and recovery, retransmits discarded TCP packet 1, and changes the size of cwnd to (cwnd/2)+3. The number of TCP packets sent by the server's TCP is now small, so the queue length of ATM-SW1 begins to decline at t2. Because CCD discards cells from the output side of the buffer, the time between t1 and t2 (T1) can be described as shown in Eq. (10).

$$T1\_ccd = 1000 \sim ((W(t1)-3) \sim Pcell \sim 53 \sim 8)/\text{input rate})) (ms) \quad (10)$$

The number of cells that accumulate in the CCD buffer during T1 (dQ(T1)) is given by

$$dQ(T1) = Nvc \sim (T1\_ccd/1000) \sim (\text{input rate-output rate})/(53 \sim 8) \text{ (cells)} \quad (11)$$

After t2, the queue length of the CCD buffer continues to decline.

After TCP packet 1 is sent the second time, the server's TCP receives the ACKs sent against packets 5 to 7. Under the conditions of TCP fast retransmission and recovery, the size of cwnd is extended by 1 packet when TCP receives one ACK, and TCP can send packets when cwnd is larger than the number of packets for which ACKs were not received by the sender's TCP. Consequently, the server's TCP sends packets 8 and 9 (see Fig. 3) when it receives the ACKs for packets 6 and 7.

After t7 when the server recieves the Ack of the retransmitted packet (packet 1), the server's TCP changes to the congestion-avoidance phase, so there is no large increase in the server's cwnd.

Figure 3 shows the changes when the parameters in Eqs. (10) to (11) are set as described above. In this case, buffer overflow does not occur, TCP timeout does not occur, and TCP throughput is high (in this model).

(b) ATM-SW1 uses EPD

As illustrated in Fig. 4, one cell from TCP packet 4 arrives at the EPD buffer of ATM-SW1 at t4. The queue length is longer than Tq, so the buffer discards all cells in incoming TCP packet 4. After receiving the third duplicate ACK, the server's TCP changes its state to fast retransmission and recovery, retransmits packet 4, and changes the size of cwnd to (cwnd/2)+3.

The buffer discards all cells in all the packets arriving at the EPD buffer when the queue length is larger than Tq. Consequently, we can represent the ratio of the number of packets discarded by EPD to the number of packets not discarded by

$$D\_epd = (\text{input rate/output rate})-1. \quad (12)$$

In this model, the value of the input rate is twice that of the output rate; therefore, Eq. (12) becomes

$$D\_epd = 1. \quad (13)$$

Consequently, packets 6, 8, 10, and 12 are discarded after packet 4 has been discarded at t1. At t2, the server's TCP changes to fast retransmission and recovery, so the queue length begins to decrease. In this model, we can represent the time between t2 and t1 (T1_epd) by

$$T1\_epd = 1000 \sim (RTT+(5 \sim Pcell \sim 53 \sim 8/\text{input rate})+(Tq \sim 53 \sim 8/\text{output rate})) \text{ (ms)}. \quad (14)$$

From the discussion in Sec. 2, whether retransmission of the dropped TCP packets during T1 is done only by fast retransmission and recovery or by using the TCP timeout is determined by the size of the server's cwnd immediately before TCP enters the fast-retransmission and fast-recovery phase and by the number of TCP packets discarded from the cwnd.

From the calculation in this section, we can determine the actions of the server's TCP and the queue length after t2. For the case shown in Fig. 4, five packets sent from a server whose cwnd is smaller than 20 packets are discarded. Therefore, TCP timeout occurs when dropped packets are retransmitted for the reason discussed in Sec. 2. Consequently, the queue length continues to decrease for a long time after t2.

3.4. Robustness of CCD

In section 3.3, we analyzed the procedure and effectiveness of CCD in a general network model. However, when CCD is installed in an ATM switch C many parameters representing the condition of the network will change every moment. Therefore, in this section, we discuss the robustness of TCP throughput with CCD with respect to (w.r.t.) these parameters.

_ Robustness w.r.t. parameter Nvc

As described before, CCD can be given parameters of the correct value only when it can understand the correct value of the active TCP connections. CCD can understand this by watching the VPI/VCI of every input (basic method) and output cell but this add some steps to the process of handling input and output cells in a normal FIFO queue. Now, we discuss a method of calculating the number of active VC connections that does not need as many steps but obtains a less precise answer than the basic method in our lab. Therefore, in this section we discuss the effect of the calculation precision based on the basic method.

When Nvc is larger than the actual number of active connections, Tt must be much larger than that expected. In that case, the possibility of buffer overflow or TCP timeout occurring is larger than in the "ideal model", where Nvc is equal to the actual number of active connections. On the other hand, we think that this does not help decrease TCP throughput because, in CCD, cells are dropped from the front of the buffer and the sender's TCP can recognize the cell discarding quickly.

When Nvc is smaller than the actual number of active connections, Tt must be smaller than that expected. In that case, there is a possibility of discarding three or more packets sent within the same TCP sending window. So the possibility of TCP timeout occurring is large.

When Nvc is much larger than the actual number of active connections, D is also much larger than that we expect. In this case, the number of discarded cells is larger and the possibility of TCP timeout occurring is larger than in the "ideal model" described in the section 3.2. But when D is twice as large as D in the model where the bandwidths of all the connections are equal, the number of discarded cells per connection is 2 and the possibility of these two discarded cells coming from two TCP packets is 0.3% (=100 ~1/30), when the number of cells forming one TCP packet is 30. This possibility is very low and, when the size of TCP's cwnd is over ten packets, TCP timeout does not occur in this case according to the discussion in section 2. Consequently, D must be set to over the number of cells forming one TCP packet when 2 or 3 TCP packets are discarded per connection and TCP timeout may occur. Therefore, the probability of TCP timeout is very small.

On the other hand, when Nvc is smaller than the actual number of active connections, D must be smaller than that we expect. In that case, CCD occurring simultaneously cannot rescue all the connections. In other word, cells that belong to some TCP connections cannot be discarded by CCD and the cwnd of those TCP connections becomes very large. Therefore, the probability of buffer overflow occurring during Tt is very large.

When Nvc is larger than the actual number of active connections, the decrease in TCP throughput tends to be small. On the other hand, when Nvc is smaller than the actual number of active connections, the decrease in TCP throughput tends to be large.

_ Robustness w.r.t. connection bandwidth

Whether the bandwidth of all the connections is wide or narrow, CCD can prevent cell discarding during Tt, so the throughput cannot change much. But in EPD, DFF, and TLD, the number of cell discarding per connection is large in the case of a large bandwidth, so the total bandwidth is regulated. Therefore, the number of discarded TCP packet is large and the throughput in such connections is small, from the discussion in section 2.

_ Robustness w.r.t. end-to-end link delay

When the end-to-end link delay is large, the throughput with the CCD model may fall to almost the same as that in the TLD model, because the effectiveness of CCD discarding cells from the front of the buffer is small in such a network model.

On the other hand, when the end-to-end link delay is small, the throughput with the CCD model may be larger than that in the model in which the end-to-end link delay is large. This is because the effectiveness of CCD discarding cells from the front of the buffer is large as long as buffer overflow does not occur. But in the CCD model, the probability of buffer overflow occurring is large, so the throughput of all the connections may not be greatly changed.

_ Robustness w.r.t. the TCP packet size

Even if the size of the TCP packet is changed, the throughput does not change much as long as CCD works correctly. But when the TCP packet is large, the number of useless cells discarded in the receiver's TCP layer is large, the average queue length is large, and the probability that the queue is longer than threshold is high. As a result, the throughput with the CCD model may fall a little. In addition, when the number of discarded cells is fixed, more TCP packets are discarded when packet size is small where cell discarding is done by TLD, EPD, or DFF. Therefore the probability of occurring TCP timeout in the model is also higher than in CCD model.

_ Robustness w.r.t. call occurrence frequency

When many calls occur during the time Tt when the CCD switch cannot discard cells, many cells may pour into the CCD buffer, which may overflow. But the usual TCP employs a "slow start[2]", so buffer overflow cannot occur as long as equation (15) is satisfied.

C ~RTT ~Ptcp<B - Tq, @ c(15)

where C is the number of calls occurring within 1 s, B is the buffer size (cells), Ptcp is the number of cells in one TCP packet, and Tq is CCD's threshold (cells).

When B is 20,000, Tq is 10,000, Ptcp is 30, and RTT is 10 ms, buffer overflow cannot occur if C is smaller than 33,333 (calls/s). This cannot be the reason for the reduction in TCP throughput.

4. Simulations

4.1 Model

The network model is shown in Fig. 2 We assumed that a client-server model sent an infinitely large FTP file. The model has from 10 to 100 servers and clients, with the number of servers and clients being equal. We divided the clients and servers into ten groups, with each group containing 1-10 clients and servers. The timing of a client sending a request packet was incremented by 100 ms from group to group, which is an asymmetrical congestion model. The traffic sent from all servers was concentrated into the buffer of ATM-SW1,which was equipped with CCD, EPD, TLD, or DFF. Therefore, this is where cell discard occurred. We set B (buffer size) to 3000 (Tq=1500) or 6000 (Tq=3000) cells. Parameters D and Tt were calculated from the number of active VCs connections (from Eqs. (5), (6), and (9)). The TCP packet size was 500 or 1500 bytes. The end-to-end link delay was 0 or 1 ms. The peak cell rate (PCR) of each connection was 1.5-10 Mbps. To isolate the effect of the traffic load for each cell-discard method, we changed the bandwidth between ATM-SW1 and ATM-SW2. The remaining parameters were set as described in Sec. 3.3. The output of this simulation was the effective throughputs of all connections in the TCP layer corresponding with network load. (Therefore the throughput is tend to be small when network load is small.)

Simulation 1:

In this simulation, we checked the results of the discussion in section 3.4 on the effect of bandwidth. Since the simulation model was originally an asymmetrical congestion model, the cwnd of each connection was varied. In addition we changed the PCR in this simulation. The number of combinations of servers and clients was either 10. We used four simulation models. The PCRs of all the connections were 1.5, 6, or 10 Mbps in first three simulations. In the last simulation, the PCRs were each 1.5 Mbps for four connections, 6 Mbps for three connections, and 10 Mbps for three connections. The end-to-end link delay of all the connections was 1 ms, and the TCP packet size was 1500 bytes, for all the connections. The buffer size was 3000 or 6000 cells. The results of the simulation are shown Figs. 5-1 to Fig. 5-8.

Simulation 2:

In this simulation, we checked the results of the discussion in section 3.4 on the effect of packet size. The simulation model used TCP packets, whose size was 500 bytes and B is 3000cells. The other parameters were all the same as in simulation 1. The results of the simulation are shown in Figs. 6-1 to 6-3.

Simulation 3:

In this simulation, we checked the results of the discussion in section 3.4 on the effect of end-to-end link delay. The size of TCP packets was 1500 bytes, which was the same in all the connections. In this simulation, the end-to-end link delay of all the connections was varied from 0 (=0 km) to 5 ms (=1000 km). The PCR of all the connections was 6 Mbps and B is 3000cells. The other parameters were all the same as in simulation 1. The results of the simulation are shown in Fig. 7.

Simulation 4:

In this simulation, we checked the results of the discussion in section 3.4 on the effect of the calculation precision. We set Nvc to a value that was varied from 5% to 6000% of the number of active VCs. In this model, the number of combinations between servers and clients was 50. The PCR of all the connections was 6 Mbps and B is 6000cells. The other parameters were all the same as in simulations 1. Fig. 8-1 shows the results of the simulation when the calculated Nvc was smaller (80%, 10%, 5%) than the real number of active VCs. Figure 8-2 shows the results when it was larger (130%, 160%, 300%, 600%, 3000%, 6000%).

Åi11 We also performed another simulation to check CCD’s tolerance to the number of VC connections. The number of combinations between servers and clients was varied from 10 to 100. In this model, the PCR of all the connection was 6 Mbps and and B is 6000cells , All the other parameters were the same as in simulation 1. The results of the simulation are shown in Fig. 8-3.

4.2 Results and discussion

Simulation 1:

The results of this simulation were almost the same as the discussion in section 3.4. The throughput of CCD was best in almost all of the simulation models. It was particularly good in the model where PCR was less than 6 Mbps. This is because cell-interleaving can be easily obtained in an ATM switch by lowering PCR. It is noteworthy that when the network load was 1.25 in Figs. 5-1, 5-3, and 5-5, the throughput of TLD was very large. One reason is that in TLD, the buffer can fully utilize the cell buffer, but CCD and EPD cannot. The second reason is that cell discarding by TLD happens only on limited connection repeatedly. This can be understood from the fairness index [5] of CCD, EPD, DFF, and TLD in the model described in Table 1. Therefore, the other connections can send cells without cell discarding, so the throughput of all the connections is relatively high.

Simulation 2:

From the Fig.6-1 – 6-3,when TCP packet size is

500 bytes, the throughput of all the methods is lower than that of Simulation 1 where the TCP packet size is 1500 bytes, because of the overhead of the TCP header. But the size of the decrease in CCD throughput is small as discussed in section 3.4 In the mixed model (Fig.6-4), the throughput is intermediate between in the cases in which the packet size is 1500 and 500 bytes. This is because the throughput for the former makes up for the latter.

Simulation 3:

From Fig. 7, the throughput of CCD is best regardless of RTT, and the trend of CCD's throughput is almost the same as that discussed in section 3.4. This is because when the end-to-end link delay is small there are many buffer overflows in the ATM switch's buffer, but CCD can prevent TCP timeout to some degree, as discussed in section 3.4. When the end-to-end link delay is large, the effect of discarding cells from the front of the buffer is small. Therefore, the throughput of DFF becomes small. However, the throughput of CCD is still large irrespective of the discussion in section 3.4, because of the effect of preventing cell discarding during Tt.

Simulation 4:

From the results shown in Fig. 8-1, we can say that CCD is very robust when the calculated Nvc is smaller than the actual number of active VCs, irrespective of the discussion in section 3.4. This is because in this model the time to start sending packet is varied. Therefore, the size of cwnd of all the connections in the case of CCD is varied and the ratio of cells in the switch's buffer which belong to a particular VC is varied. As a result, we can conclude that CCD discards mainly cells that belong to the connections where the size of cwnd is very large. In a real network the sizes of cwnds are always varied and the behavior of the throughput applies to this simulation model. But in some rare cases where the cwnd of all the active connections is the same, the throughput of CCD is smaller than that in this simulation, as discussed in section 3.4.

From the results shown in Fig. 8-2, we can say that CCD is very robust when the calculated Nvc is much larger than the actual number of active VCs, as discussed in section 3.4. When Nvc is 3000% of the active VCs, CCD brings about 2 or 3-packet discarding, which may leads to TCP timeout. Therefore, the throughput of CCD is reduced in some cases. When CCD is 6000%, it brings about 3-packet discarding, which almost leads to TCP timeout. Therefore, the throughput is worse than EPD in every case. But overall, this simulation proves CCD's robustness with respect to Nvc.

From the results shown in Fig. 8-3, we can say that the throughput of CCD is best regardless of the number of active connections.

5. Conclusion

We have proposed a new method of discarding cells when an ATM switch becomes congested. To prevent TCP timeout, cell discarding is suspended for a specified time. To prevent buffer overflow while cell discarding is suspended, the queue length is minimized by outputting the packets following the discarded packet quickly; this is done by discarding cells from the front of the buffer.

Through calculation and simulation, we found that using CCD results in higher throughput than using an ordinary method of cell discarding such as TLD, DFF, or EPD in a TCP-over-UBR network even though CCD does not consider TCP packet size or boundaries. In addition, CCD is robust against changes in the any congestion models.

References

[1] K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP", Computer Communications Review, June 1996.

[2] W. R. Stevens, "TCP/IP Illustrated, Volume 1", Addison Wesley, p. 299, March 1996.

[3] A. Romanow and S. Floyd, "Dynamics of TCP Traffic over ATM Networks", Proc. ACM SIGCOMM 1994, pp. 79-88.

[4] T. V. Lakshman, "The Drop From Front Strategy in TCP and in TCP over ATM", Proc. INFOCOM 1996, pp. 1242-1250.

[5] ATM Forum, "Baseline Text for Traffic Management SWG", ATM-Forum/94-0394R5.

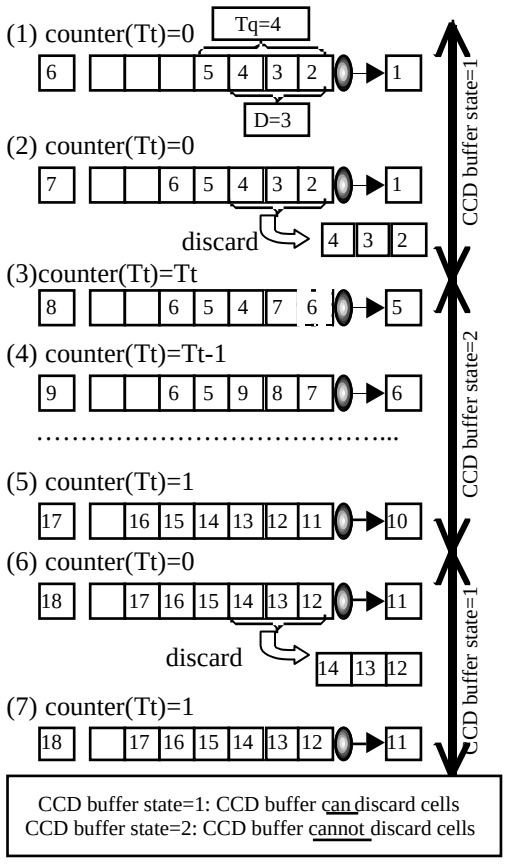

Fig.1: Proposed CCD method

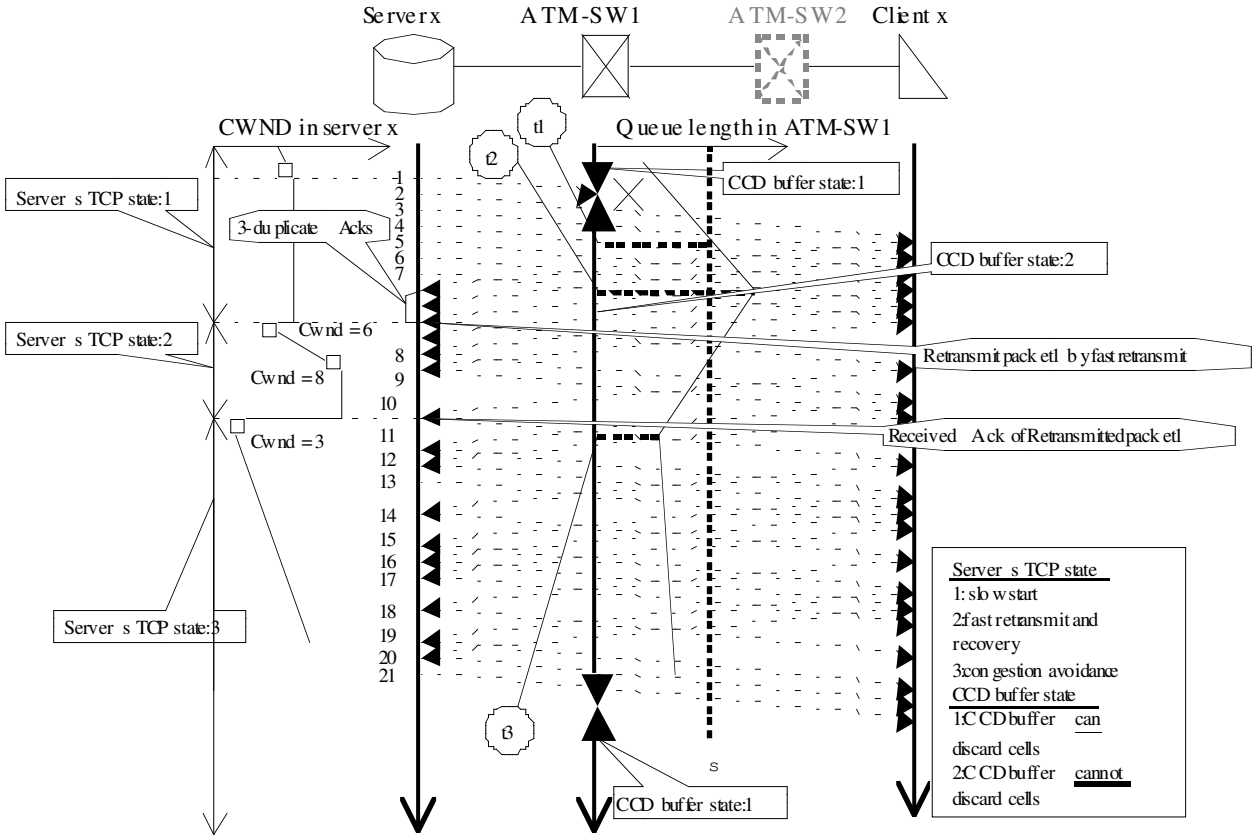

Fig.3: Example behavior of server s TCP and resulting change of Queue length of CCD buffer.

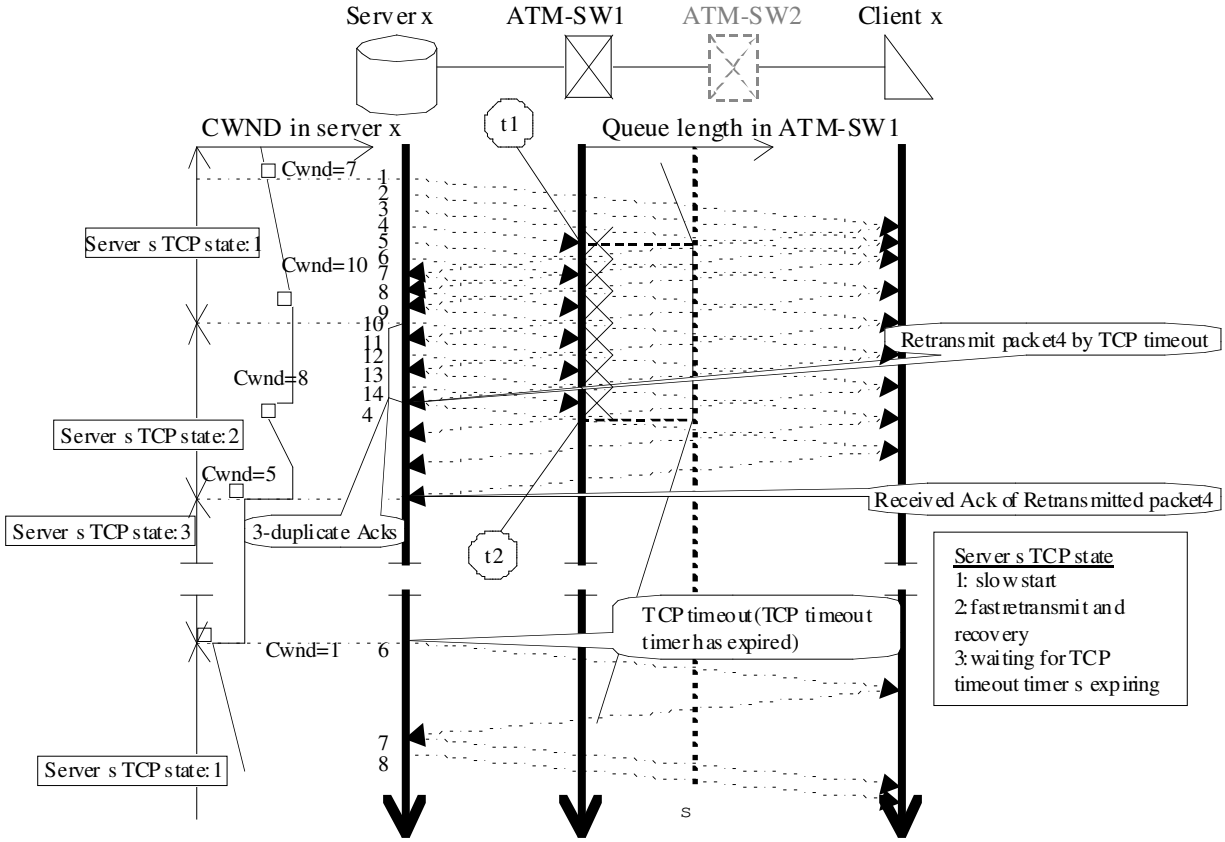

Fig. 4: Example behavior of server s TCP and resulting change of Queue length of EPD buffer.

	CCD	EPD	DFF	TLD
Fig.6-1	0.99	0.84	0.90	0.86
Fig.6-3	0.95	0.96	0.99	0.87
Fig.6-5	0.97	0.95	0.95	0.85

Table1:fairness index

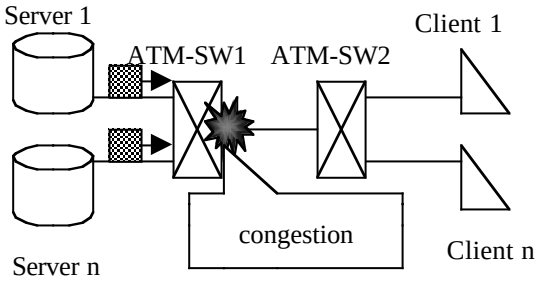

Fig. 2: Network model of TCP over ATM

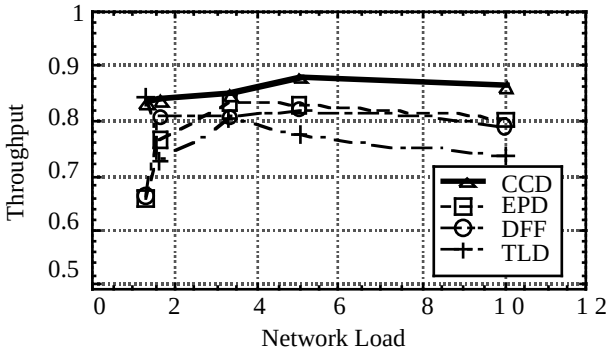

Fig.5-1 PCR1.5Mbps and B=3000

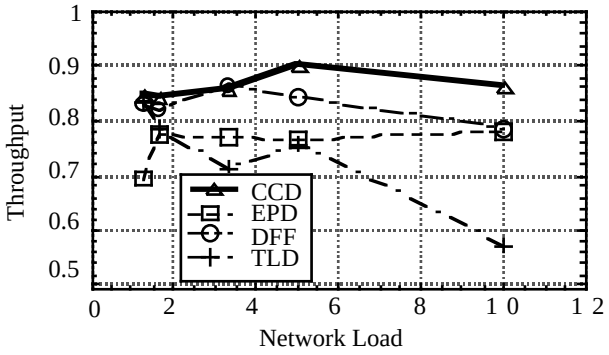

Fig.5-2 PCR1.5Mbps and B=6000

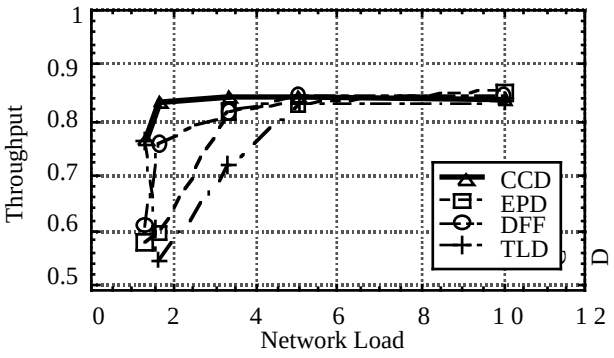

Fig.5-3 PCR6Mbps and B=3000

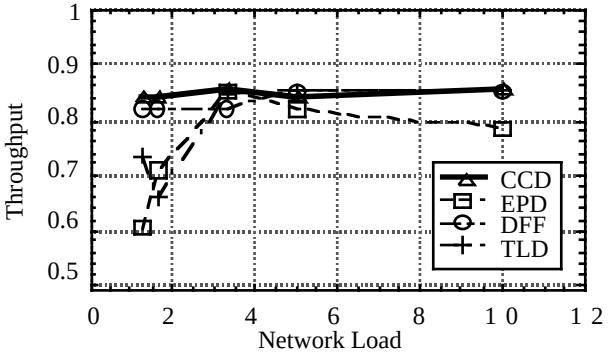

Fig.5-4 PCR6Mbps and B=6000

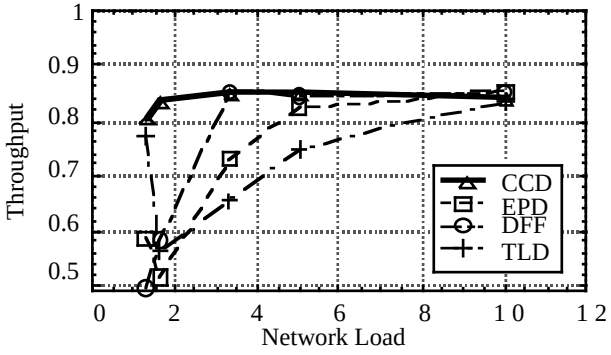

Fig.5-5 PCR10Mbps and B=3000

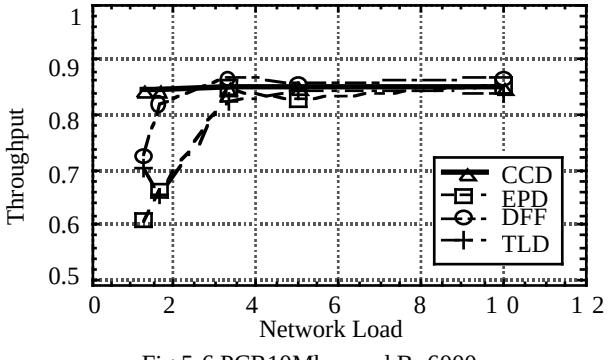

Fig.5-6 PCR10Mbps and B=6000

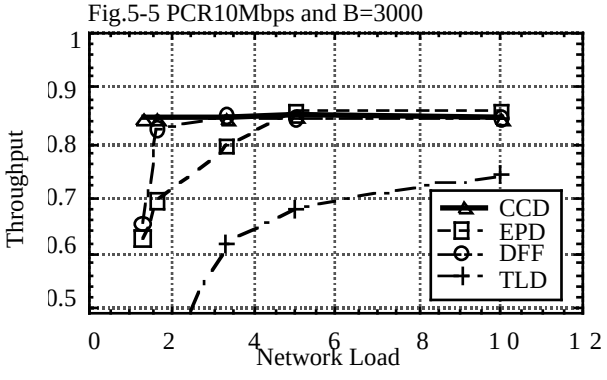

Fig.5-7 mixed PCR B=3000

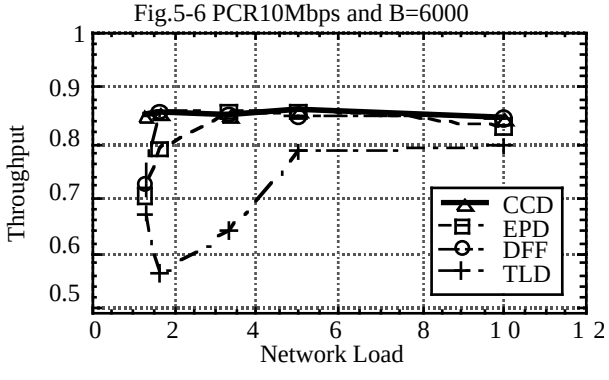

Fig.5-8 mixed PCR B=6000

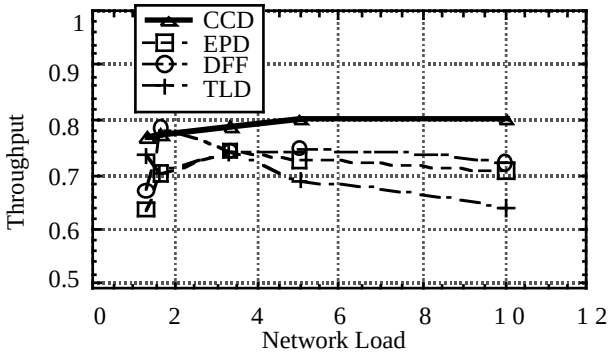

Fig.6-1 PCR1.5Mbps

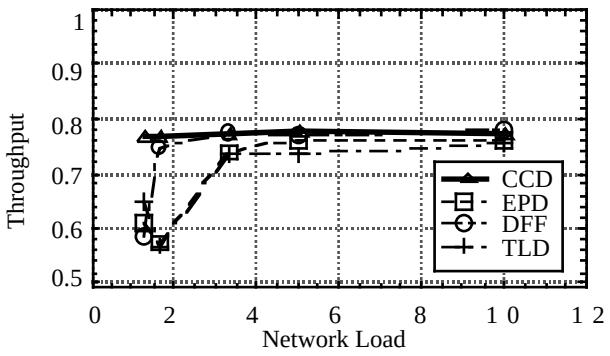

Fig.6-2 PCR6Mbps

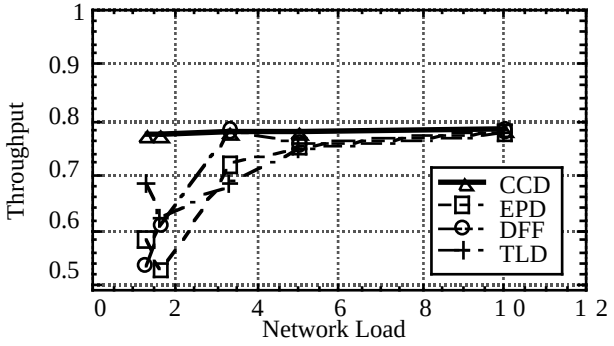

Fig.6-3 PCR10Mbps

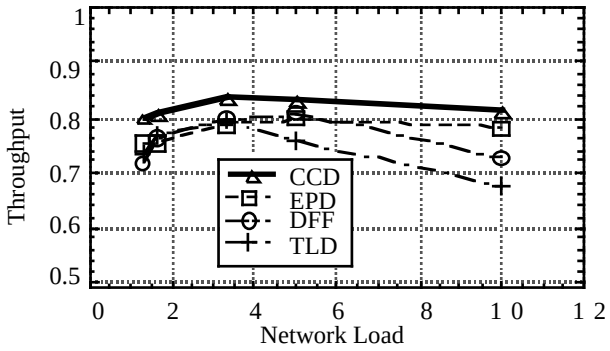

Fig.6-4 mixed Packet Size PCR=1.5Mbps

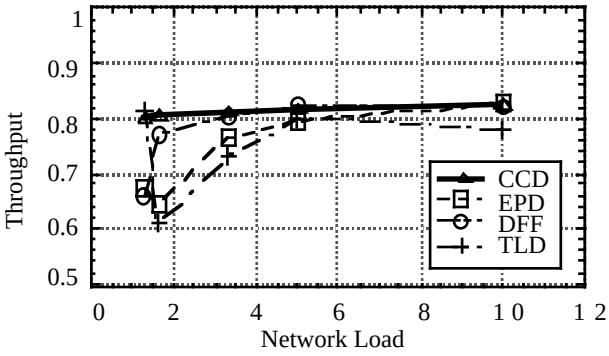

Fig.6-5 mixed Packet Size PCR=6Mbps

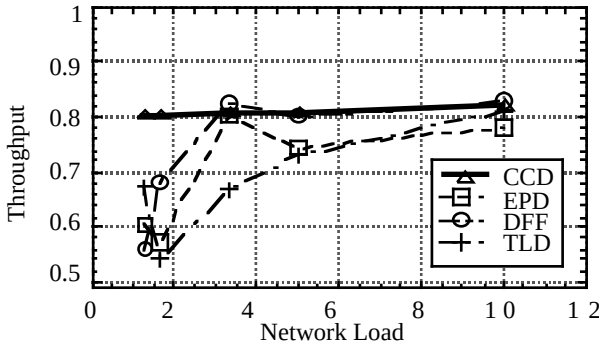

Fig.6-6 mixed Packet Size PCR=10Mbps

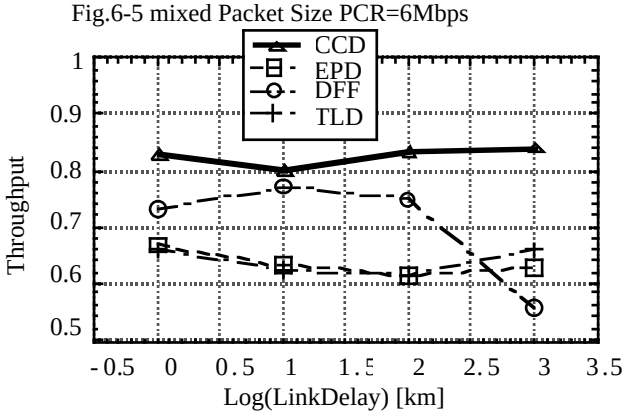

Fig.7 effect of end to end link delay(PCR=6Mbps)

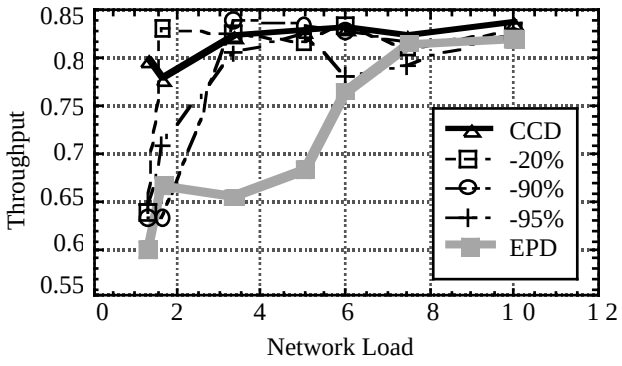

Fig.8-1 effect of the preciseness of the calculation

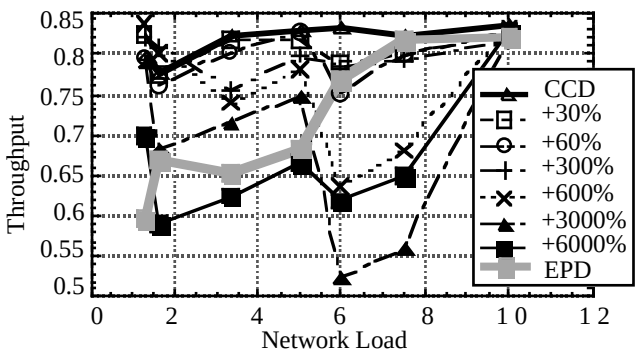

Fig.8-2 effect of the preciseness of the calculation

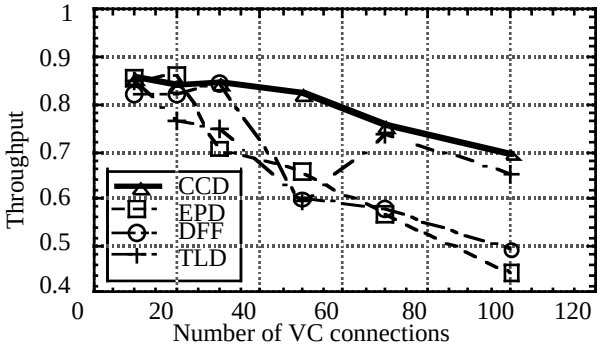

Fig.8-3 effect of number of VC connections