IEICE TRANS. COMMUN., VOL.E90–B, NO.8 AUGUST 2007

PAPER

A Cross-Layer Retransmission Control for Improving TCP Performance in Wireless LAN

Satoshi OHZAHATA[†a)], Shigetomo KIMURA[††], Yoshihiko EBIHARA[††], *and* Konosuke KAWASHIMA[†], *Members*

SUMMARY In this paper we propose a cross-layer retransmission control for TCP communication over a wireless link. With our proposed control, a retransmission delay for lost packet is reduced, packet losses in the wireless link are eliminated and all packets are delivered in the correct order. No change is required to TCP itself or to the sender. Our proposed method is implemented in a queue between the media access control (MAC) layer and logical link layer in a base station, and is designed to assist local retransmission control in the MAC layer. Computer simulations show that our proposed method can maximally use the bandwidth of the wireless link under high bit error rates conditions with conventional TCP control. The fairness problem of TCP communication between connections with different bit error rates in a wireless link is also improved, and MAC level fairness is also controllable.

key words: *wireless TCP, cross-layer, queue management, retransmission control, IEEE 802.11*

1. Introduction

Wireless Internet environments have come into wide use, since they can be constructed more easily and flexibly than wired Internet installations nowadays. In the Internet, reliable data communication is mainly controlled by TCP [1]. However, congestion control is also necessary when packet loss occurs due to a wireless link error, and this always decreases the goodput of the connection. To improve TCP performance in wireless environments, many methods have been proposed to adapt congestion control to the wireless environment [2]–[9]. These end-to-end control schemes improve the performance for random or sporadic packet losses, but an end-to-end control of TCP takes the round-trip time to complete the control. Thus, the improvement of performance is smaller than that of modified base station (BS) methods below.

In the snoop method [10], the snoop agent works at the Logical Link (LL) layer in a BS and monitors the packet loss in the wireless link by checking the sequence number of the TCP header for every packet. If the agent detects a loss, the lost packet is locally retransmitted from the cache in the agent to completely eliminate the packet loss. To improve the performance further, the agent removes the mis-ordered packets and sends these packets in the correct order not to produce extra duplicate ACK by the receiver. However, these controls involve an additional delay time for the feedback control of TCP. In addition, these controls sometimes break the semantics of TCP congestion control, which is originally operated by the sender and receiver.

In explicit loss notification (ELN) [11], BS detects a packet loss in the wireless link, and then explicitly notifies the fact to the sender. Based on the notification, the sender can select a suitable congestion control corresponding to the reason for the loss. In a technique described in [12], ELN is introduced to the snoop mechanism, to notify the sender of the occurrence of a packet loss in the wireless link. The transport unaware link improvement protocol (TULIP) [13] is almost the same with Snoop method, but improves the TCP goodput a little and also the round-trip delay compared with the snoop method because TULIP has the MAC acceleration to reduce the local feedback delay of ACK from the receiver.

These methods improve TCP performance in wireless Internet environments, but also require complicated control and significant modifications to the conventional systems. Furthermore, many methods ([1]–[13]) do not well consider local retransmission control in the MAC layer in spite of the fact that many kinds of wireless MAC media have such retransmission control for reliable communications. The packet loss recovery concept in LL layer ([10]–[13]) is effective when MAC retransmission control does not recover packet losses but this control also arises out of order packet in TCP level. To remove these packets, additional delay and complicated control in TCP level is required.

To solve the delay and complicated control problems, we propose a cross-layer retransmission control between LL and MAC. Our proposed control can reduce the packet retransmission delay until one turn of packet scheduling between LL layer and MAC layer in a node. Our retransmission control assists MAC retransmission control to serve no packet loss in wireless link without affecting the other flows. With this control, packets are also delivered in the correct order over an IEEE 802.11 wireless link. The sender never receives a duplicate ACK from the receiver due to the packet loss in the wireless link error. Then, we do not modify TCP itself and the sequence number information in TCP and LL headers are never referred to. The TCP agent of the sender can control the traffic to the wireless terminal as if the terminal were connected with a wired link. We also show that

Manuscript received July 18, 2006.
Manuscript revised February 7, 2007.

[†]The authors are with the Division of Systems and Information Sciences, Tokyo University of Agriculture and Technology, Koganei-shi, 184-8588 Japan.
[††]The authors are with Institute of Information Sciences and Electronics, University of Tsukuba, Tsukuba-shi, 305-8573 Japan.
a) E-mail: ohzahata@cc.tuat.ac.jp

DOI: 10.1093/ietcom/e90–b.8.2070

our method achieves the maximum or almost the maximum goodput even in the high bit error rate (BER) conditions of a wireless link with maintaining fairness for each flow in TCP level. MAC level fairness is also provided by adding information of MAC level throughput to the queue management method. These evaluations are given by computer simulation experiments.

The rest of this paper is structured as follows. Section 2 describes our proposed cross-layer retransmission control. In Sect. 3, the computer simulation experiments are described. Finally, Sect. 4 provides conclusions.

2. Proposed Methods

2.1 Proposed Queue Management Method for Cross-layer Retransmission Control

To improve TCP performance in wireless environments, we consider the following things. No packet loss is allowed. No extra duplicate ACK is produced by the receiver. All data segments should be delivered in the correct order. No variation of the round-trip time and no additional the round-trips for packet recovery are required. Our proposed method is designed to satisfy these requirements below.

Design Policies of Our Proposed Method:

1. Our proposed method is specialized for IEEE802.11.
2. We do not modify the TCP congestion control algorithm.
3. To simplify the control and add no extra delay, we do not manage the sequence numbers of LL and TCP.
4. The local retransmission mechanism in MAC is enhanced to execute the retransmission control, to serve a no-packet-loss link.
5. However, although this control is executed until each packet is received correctly at the terminal, it is not permitted for a particular packet in a particular flow to occupy an excessive amount of bandwidth, to the detriment of other flows.
6. Our modifications are implemented in the queue management in the LL and MAC, and no change is required for the higher layers.
7. A queue is prepared for a terminal of TCP flows. If a common queue is used for all terminals, a packet sent to a terminal under extremely poor radio conditions may prevent the transmission of packets to the other terminals. TCP flows are identified by Protocol or Service Level Agreement field in IP header.

Figure 1 shows procedures of our proposed method. A queue is provided between the MAC layer and LL layer for each terminal in the BS. These queues are managed by a scheduling control which is descried in the next subsection. Our proposed method is realized by the following procedures.

When one of the queues is allowed to send a packet, (1) the head packet in the queue is copied to the MAC layer for transmission, but the original remains at the head of the queue. In the next step, (2) the MAC layer forwards the packet to the receiver. If the transmission is successful, (3) the acknowledgment is notified from the MAC layer in the receiver, and (4) is also signaled to the queue, to delete the packet from the head of the queue. In this case, the next queue is allowed to transmit a packet.

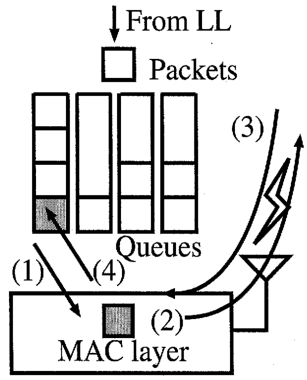


Fig. 1 Steps in proposed queue management procedure with cross-layer retransmission in a BS.

Otherwise, the MAC layer continues retransmitting the packet until acknowledgment is received by usual IEEE 802.11 MAC control. If the local retransmission cannot recover the packet loss, i.e., the acknowledgment shown in Fig. 1 (3) cannot be received, the retransmission attempts will exceed the predefined maximum number. In this case, the packet is eliminated from the MAC layer, and the next queue is allowed to send a packet. Since the eliminated packet remains at the head of the queue, it can be retransmitted to the receiver again when the queue has its next turn to send and the retransmission delay can be reduced until this small duration.

It should also be emphasized that the packet order in a flow is never changed as a result of wireless link error in the above procedures because IEEE 802.11 MAC adopts a "stop and wait protocol." Thus, even when the error rate on the wireless link is much higher, the TCP agent in the sender recognizes by the error that the wireless link momentarily has a narrower bandwidth, and that a long round-trip time is required, and the TCP congestion control adequately adjusts the flow. All ACK packets are immediately delivered to the sender since any modification is added to the intermediate nodes. To avoid indefinite retransmission in MAC, when two times of RTS/CTS exchanges are failed in sequence, the head of queue is discarded because the packet loss let the sender know the network status.

2.2 Packet Scheduling Control

How to share bandwidth between no-error terminal and error-prone terminal is important because fairness in IP level and MAC level is different when MAC retransmissions for packet error occur in wireless link. We propose two packet scheduling controls to serve fairness in IP or MAC level.

2.2.1 Consideration of Fairness in IP Level (Proposed Method 1)

Both of our proposed methods are implemented by an extension of Deficit Round Robin (DRR) [14]. In DRR, a queue is prepared by a terminal and the bandwidth is fairly shared among terminals by "throughput" rather than "the number of packets" when the packet is sent to MAC layer. A "quantum," which is a fixed size of bytes, is given for each terminal in each round. If the cumulative quantum for a queue is larger than the packet size of packet of the head of queue, the packet is delivered to the MAC layer at the round, and then the cumulative quantum is reduced by "the packet size." In the case of that the buffer for the queues is full, the last packet of the largest size of queue is removed.

By above procedures, IP level fairness is provided even if one of the TCP flow is delivered to error-prone terminal. This is because that our proposed cross-layer retransmission control can prepare no-packet-loss link for the higher layer, the bandwidth for each terminal is shared by the outgoing bandwidth from the queue. Then, all flows have to wait the same period until the next tern of packet transmission to MAC layer by this packet scheduling. We call this method "Proposed method 1" in the following [15].

2.2.2 Consideration of Fairness in MAC Level (Proposed Method 2)

In the case of that the MAC retransmissions occur many times, shared bandwidth in IP level and MAC level is different in Proposed method 1. In this case, MAC level bandwidth is much more consumed for a terminal with error-prone wireless link and this reduces goodput and requires additional delay for a terminal with no error one. Then, we have to consider the number of MAC retransmissions when the outgoing bandwidth is shared among the terminals.

This control is realized by recalculating the cumulative quantum when the MAC retransmission is executed. In "Proposed method 2," "the size of packet × the retransmission counts in MAC layer" is additionally reduced from the cumulative quantum when the packet is transmitted successfully, or the number of MAC retransmission count is exceeded. This control enables to consider used MAC bandwidth for each terminal when it sends the packets to MAC layer. The number of retransmission counts for the head of packet is informed by procedure (4) in Fig. 1 and the quantum is calculated.

With above procedures, Proposed method 2 can give MAC layer fairness because outgoing packet from the queue is adjusted for error-prone terminal, and the number of outgoing frames from the BS is fairly shared between error-terminal and no-error terminal. In addition, effect of retransmission from error-prone terminal is reduced by the shared MAC bandwidth for no-error terminal, and the delay for no-error terminal is also improved.

3. Simulation Experiments

To evaluate our proposed methods, computer simulation experiments are executed using the network simulator ns-2.1b8 [16]. We show in the following experiments that our proposed methods achieve maximum or almost maximum throughput for MAC level even in high bit-error link conditions. In addition, our methods fairly share the bandwidth in IP or MAC level. To clearly evaluate the results, we compare four TCP communication methods. The first and second methods are our Proposed methods 1 and 2. The third queue management method is original DRR. The fourth one is the snoop method [10], in which a FIFO queue is used between the MAC layer and LL layer in the BS. The snoop cache is also included in the FIFO queue. The network model is depicted in Fig. 2. Simulation results satisfy that 95% confidence interval is smaller than 5% of the average value. Common conditions for all simulation experiments are as follows.

Queue conditions: In the first, second and third methods, the BS prepare a queue, whose management method is DRR, for each terminal. A size of quantum is 250 octets. All queues in the four methods are constructed with a shared memory, and the total memory size to store packets is 150000 bytes (100 packets) as the default. The maximum size of the snoop cache is the same as with the total buffer size. All of the methods are implemented only in the BS.

Network conditions: The transfer speed of each link is depicted in Fig. 2. The wired link delay between the sender and the BS is set to either 1 ms or 100 ms. The distance between the BS and each terminal is fixed at 1 m. For a wireless link between the BS and each terminal, the BER in the physical layer is selected over the range 1×10^{-6} to 5×10^{-5}. Medium access control of the wireless network is IEEE 802.11 implemented in ns [16].

Traffic conditions: New Reno version of TCP is used. For each terminal, the sender prepares a source which continuously sends packets to the terminal by FTP under TCP congestion control of New Reno. The sizes of a TCP data seg-

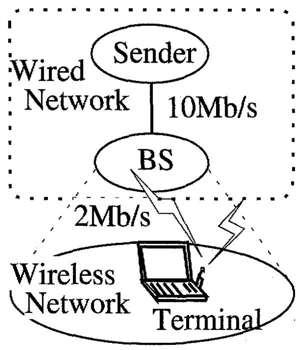


Fig. 2 Simulation model.

ment and an ACK are fixed at 1460 octets and 40 octets, respectively. The maximum congestion window size of TCP is 20 segments (29200 octets).

3.1 One Terminal Model (Proposed Method 1)

3.1.1 Average Goodput Evaluation

In the first simulation experiment, one terminal exists in the network. Figures 3 and 4 show the average goodput in TCP with the wired link delay of 1 ms and 100 ms, respectively. All packet losses arise in the wireless link, because the size of each buffer in the BS is large enough to avoid overflow. "Proposed 1" means our proposed method 1, and "Snoop" or "Original" is that the queue management method is the snoop method (mentioned in Sect. 1) [10] or original DRR method. Since there is no difference between Proposed methods 1 and 2 in this model, we only show results of Proposed method 1 in this subsection.

From the figures, all methods show a lower average goodput as the BER becomes higher. Proposed method 1 improves the average goodput for the snoop method by as much as 30% under in each wired link delay condition. With a wired link delay of 100 ms, the packet loss penalty is larger, and the average goodput of Proposed method 1 is clearly better than the snoop method and the original method at higher BERs.

In the snoop method, the terminal receives some packets in the wrong order. Then the snoop agent should remove the duplicate ACKs from the terminal to avoid unnecessary congestion control in the sender. Therefore, when the wireless link error rate is quite high, many of duplicate ACKs are discarded. Indeed, 70% of ACKs are discarded by the snoop agent in high BER cases. This fact breaks the feedback control of TCP and the sliding window control mechanism does not work well.

Figures 5 and 6 show the average MAC throughput between the BS and terminal for wired link delays of 1 ms and 100 ms, respectively. The MAC throughput means the throughput of outgoing data frames at the MAC level from the BS or terminal. Thus, these graphs show how each method effectively uses the bandwidth at the MAC layer in the wireless link. "Total" means the average MAC throughput between the BS and terminal. "From BS" and "From Terminal" are the average MAC throughput from the BS to the terminal and vice versa.

Even under a higher BER, our method keeps or increases the "Total" MAC throughput. In the case of 1 ms delay, "Total" keeps the throughput. In the higher BER case, "From BS" increases the bandwidth because of the retrans-

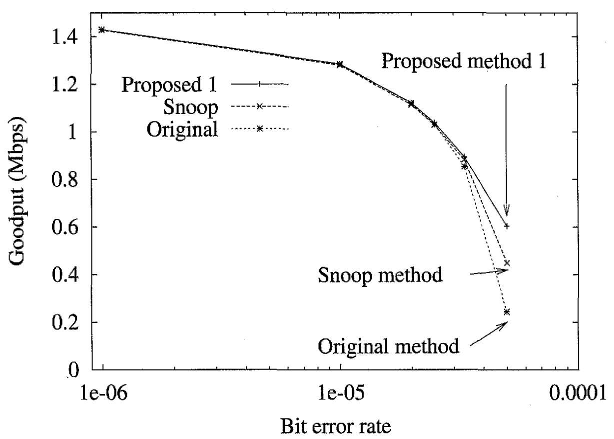


Fig. 3 Average goodput of TCP (delay 1 ms).

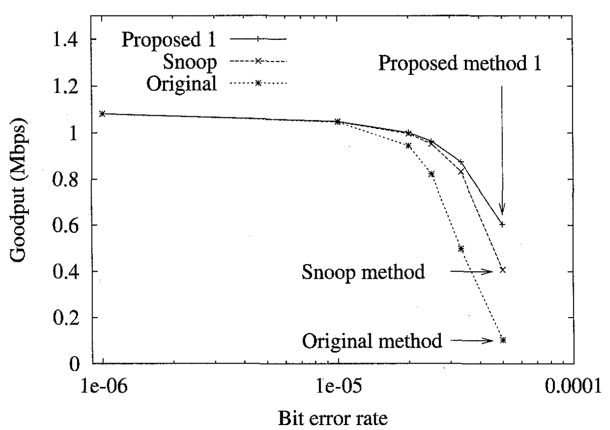


Fig. 4 Average goodput of TCP (delay 100 ms).

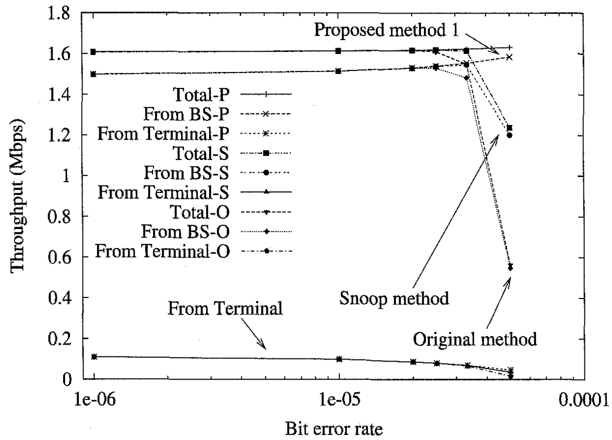


Fig. 5 Average MAC throughput (delay 1 ms).

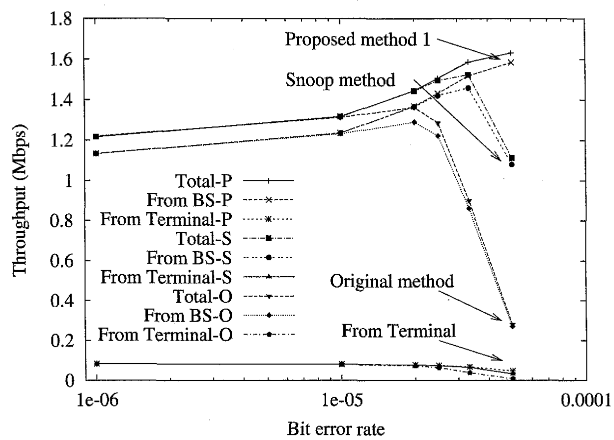


Fig. 6 Average MAC throughput (delay 100 ms).

mission of the data frames of TCP data. The data frames of TCP ACK "From the terminal" is reduced, but "Total" is slightly increased for the high BER condition. The "Total" MAC throughput is almost same as with the maximum throughput of IEEE 802.11 with a traffic load of 1.0 [17], and the MAC bandwidth is always used. Thus, the goodput in Fig. 3 cannot increase any further in the high BER case. For 100 ms delay, "From BS" and "Total" are increased for the high BER case. This means that the MAC layer is much more effectively used than with a lower BER in spite of the fact that many wireless link errors occur, because so many times MAC retransmissions need the extra bandwidth to recover the lost frame.

3.1.2 Average Queue Length Evaluation

Next, we discuss the queue length in the BS for each method. Figures 7 and 8 show the average queue length in the BS when the wired link delay is 1 ms and 100 ms, respectively. Note that the total queue length is sampled every 100 ms to obtain the average queue length. In Proposed method 1, the queue length increases for high BER in both link delay conditions, because the retransmission of packets from the queue to the MAC layer occurs many times, and the control makes packets stay in the queue longer. In each link delay, the original method reaches a peak when the BER is 2×10^{-5} or 2.5×10^{-5}. Te local retransmission covers the packet loss, then TCP does not need to retransmit packets or reduce the congestion window, and the packets in the queue must wait longer until the local retransmission is completed. In higher BER conditions, the local retransmission cannot cover the packet loss and the congestion window tends to be small, and then the queue length becomes short. In the snoop method, the resulting lengths stay between those of Proposed method 1 and the original one. This is because the snoop recovers the wireless packet loss but breakage of the TCP semantics reduces the performance.

3.1.3 Average Round Trip Time Evaluation

Figures 9 and 10 show average round trip time (RTT) from sending a data segment to receiving its ACK segment. In determining the average, only the times of successfully delivered segments are considered, and packets lost in the wireless link are ignored since the delay of such packets is difficult to define. In the lower BER conditions, the three methods produce the same results, but in the higher BER cases, our method and the snoop method increase the delay and the original method reduces it (Fig. 9). The increment of

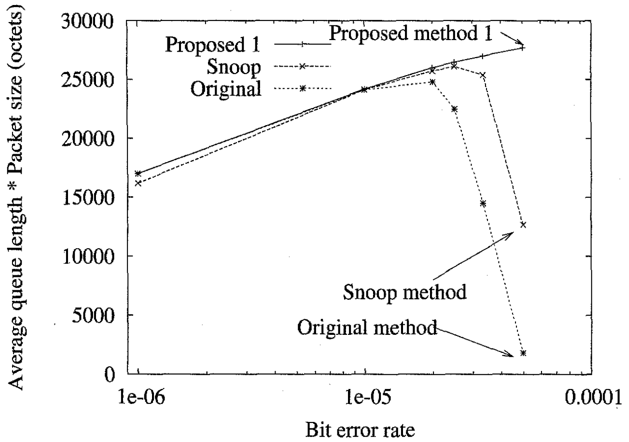


Fig. 7 Average queue length in BS (delay 1 ms).

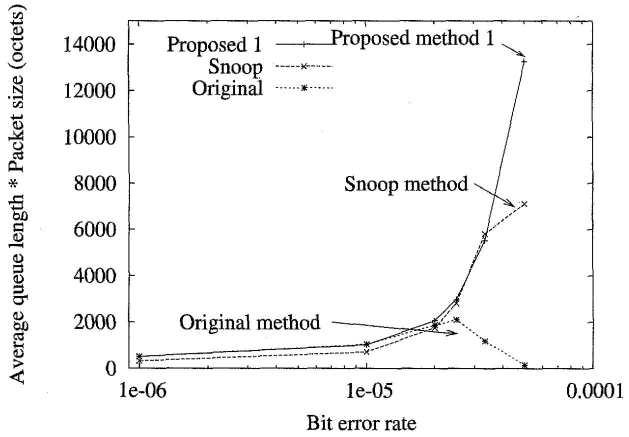


Fig. 8 Average queue length in BS (delay 100 ms).

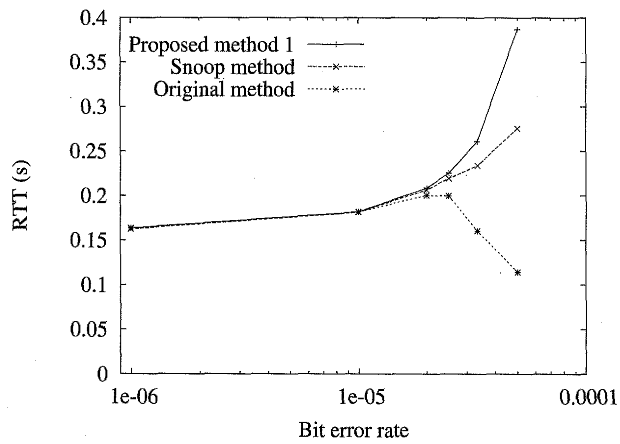


Fig. 9 Average round trip time (delay 1 ms).

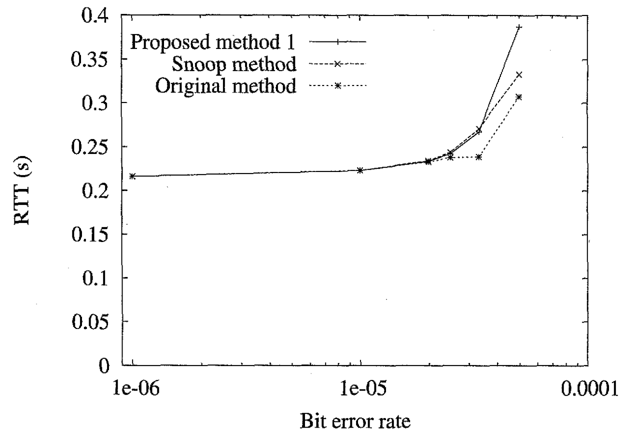


Fig. 10 Average round trip time (delay 100 ms).

the delay comes from queuing delay in the BS, since the lost packet is delivered from the head of the queue in our method, or the snoop caches until the terminal receives the packet correctly. In the snoop method, the TCP sender reduces the congestion window and there is no congestion in the networks, which results in a smaller RTT than our method. RTT of the original method is smaller because the average RTT is only the mean of successfully delivered packets that have no queuing delay in the BS, as seen in Figs. 7 and 8. In Fig. 10, all the methods increase RTT at higher BER. In this condition, RTT includes at least 200 ms and additional queuing delay. This fact explains the similar results for the three methods.

3.1.4 Influence of Limited Queue Buffer Size

In order to examine the influence of limited queue memory on Proposed method 1, Figs. 11 and 12 indicate results of the goodput for the three methods when the total buffer size in the BS is limited to 20000 and 150000 bytes respectively. As in Fig. 7, the packet loss comes from congestion in the wired network and link errors in the wireless network.

From Fig. 11, Proposed method 1 and the original method give almost the same results for each buffer size condition because RTT is small, so the congestion control quickly works at the sender. The snoop method, however, reduces the goodput when the BER is lower than or equal to 1×10^{-5}. The snoop cache is prepared for the unique queue between the LL and MAC in the BS, and the cached packets are not removed until the ACK is received by the snoop agent. Therefore, the queue is relatively smaller than that of the other two methods. However, with a high BER, the average goodput under the limited buffer size is higher than that in the unlimited case. When a packet is lost in a wired network, since the snoop agent in the BS does not remove a duplicate ACK, the sender can receive the ACK from the receiver without the unnecessary control. Indeed, when the BER is 5×10^{-5}, 50% of ACKs are returned to the sender from the receiver. Note that reference [12] also discusses the size limitation of the snoop cache from the other point of view.

From Fig. 12, Proposed method 1 and the snoop method reduce the goodput with a limited buffer size. Since RTT is long, the congestion control of TCP does not work immediately. Proposed method 1, however, can retain higher goodput than the snoop method, even when packets are lost at both by network congestion and the wireless link at nearly the same time.

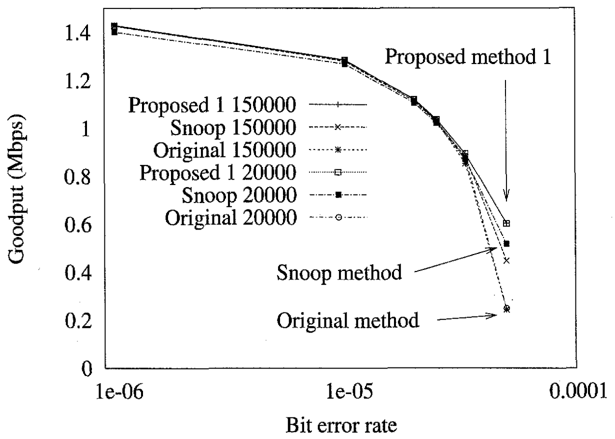


Fig. 11 Average goodput with limited buffer size (delay 1 ms).

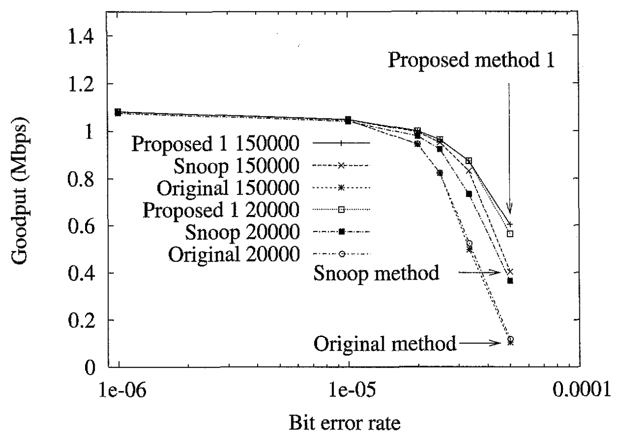


Fig. 12 Average goodput with limited buffer size (delay 100 ms).

3.2 Two Terminals Model

In this subsection, we show influences of two TCP connections over a wireless link.

3.2.1 Average Goodput Evaluation

Figures 13–16 depict the average goodput of the four methods consisting of two terminals with a wired link delay of 1 ms. The BER of one terminal is fixed at 1×10^{-6} (no error terminal), while that of the other terminal is changed from 1×10^{-6} to 5×10^{-5} (error terminal). When the BER is less than or equal to 1×10^{-5}, the two terminals fairly share the bandwidth of the wireless link for the four methods. The local retransmission control in IEEE 802.11 effectively works for the packet loss.

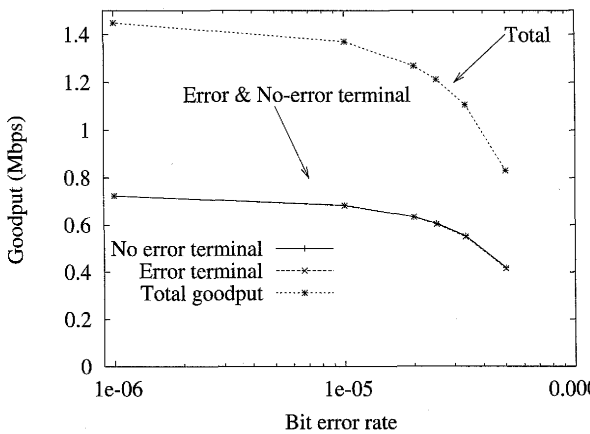


Fig. 13 Average goodput of Proposed method 1 when the BERs of two terminals are different (delay 1 ms).

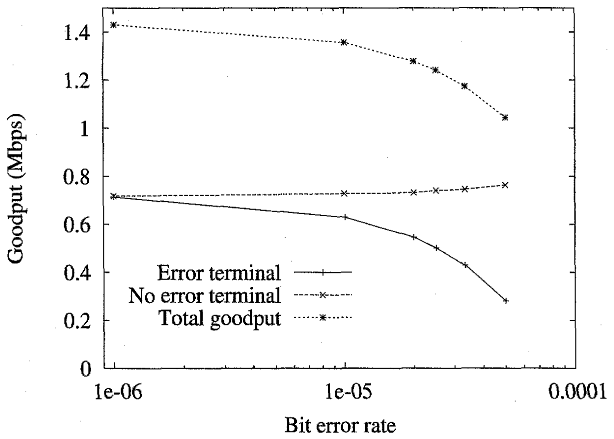


Fig. 14 Average goodput of Proposed method 2 when the BERs of two terminals are different (delay 1 ms).

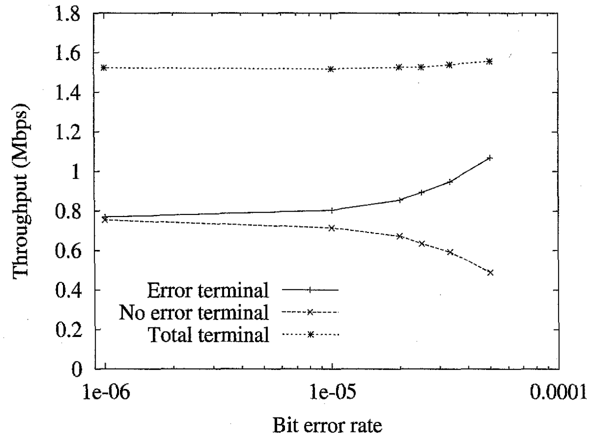


Fig. 17 Average MAC level throughput of Proposed method 1 when the BERs of two terminals are different (delay 1 ms).

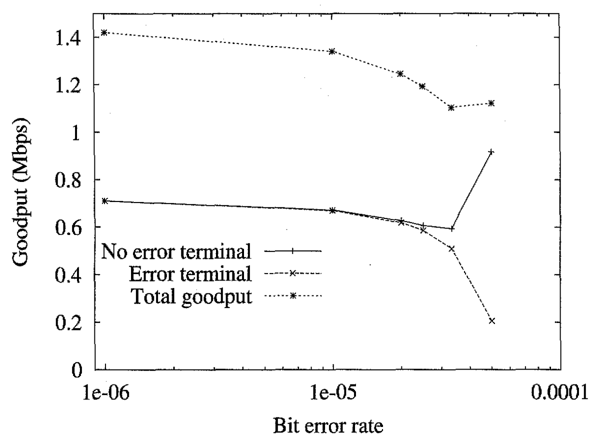


Fig. 15 Average goodput of the snoop method when the BERs of two terminals are different (delay 1 ms).

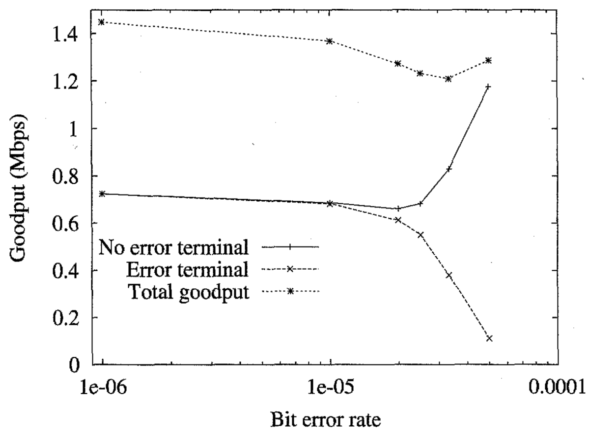


Fig. 16 Average goodput of the original method when the BERs of two terminals are different (delay 1 ms).

In Proposed method 1, the bandwidth is also fairly shared even at high BERs, because both terminals can receive packets in the correct order without any lost packets and have almost the same end-to-end delay. Then the difference in the link condition does not interfere with the end-to-end control at the sender. However, the average of the total goodput at the high BER decreases 40% from the maximum. The reduction in the bandwidth is used to retransmit packets at the MAC layer for the error-prone terminal.

In Proposed method 2, TCP goodput is only reduced for the error terminal because MAC bandwidth is fairly shared between the two terminals by the scheduling control (see Fig. 18). The reduced bandwidth for higher BER condition is used as retransmissions for the packets loss of error terminal. However, goodput of no error terminal is not affected by the error prone terminal even when the error terminal is in heavily wireless error conditions.

However, in the snoop method and original method, the average goodput for the high BER terminal is significantly reduced. Then, the no error terminal obtains the reduced bandwidth from the high BER terminal, which causes unfairness between the two flows. In the snoop method, if the retransmission controls in MAC loses a packet in the wireless link, the LL level retransmission recovers the loss. The LL level retransmission, however, is not executed until the snoop timeout occurs or a duplicate ACK arrives at the snoop agent. This delay makes RTT longer for only the error-prone terminal. In the original method, the wireless loss should be recovered by the end-to-end control of TCP, and the delay becomes large and the performance becomes much worse.

Figures 17–20 show throughput in MAC layer. In Proposed method 1, the error terminal consumes larger MAC bandwidth than that of the no error terminal. The increased bandwidth for error terminal is used for MAC retransmissions and the bandwidth for no error terminal is reduced. To keep goodput in TCP level, larger MAC level bandwidth should be prepared for the error terminal.

Proposed method 2 can fairly share bandwidth in MAC layer for every wireless link condition because the scheduling control considers the MAC retransmissions when it shares the bandwidth. Since the bandwidth in MAC layer is appropriately prepared for the both terminals, communication quality for the no error terminal is not affected by the error terminal in every wireless condition.

In Snoop and Original methods, the error terminal consumes larger bandwidth for low error bit error condition.

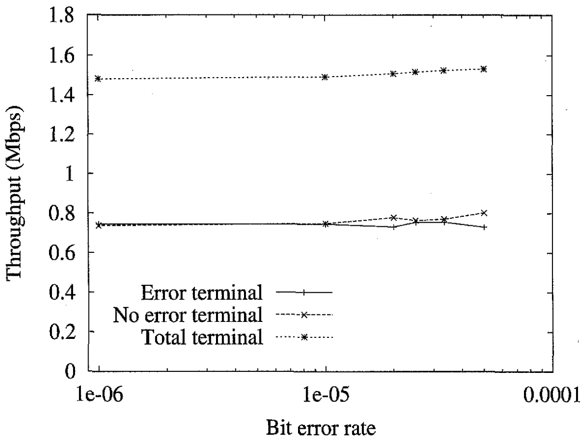


Fig. 18 Average MAC level throughput of Proposed method 2 when the BERs of two terminals are different (delay 1 ms).

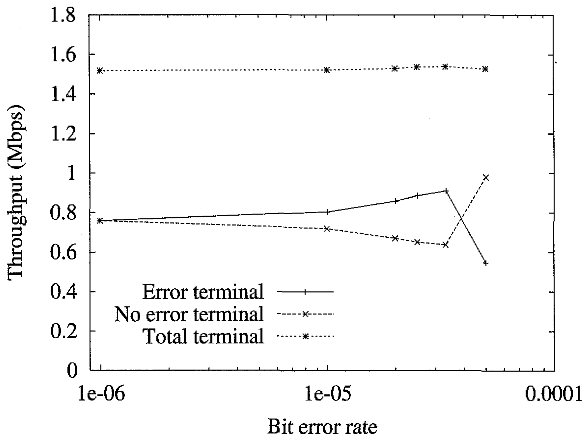


Fig. 19 Average MAC level throughput of Snoop method when the BERs of two terminals are different (delay 1 ms).

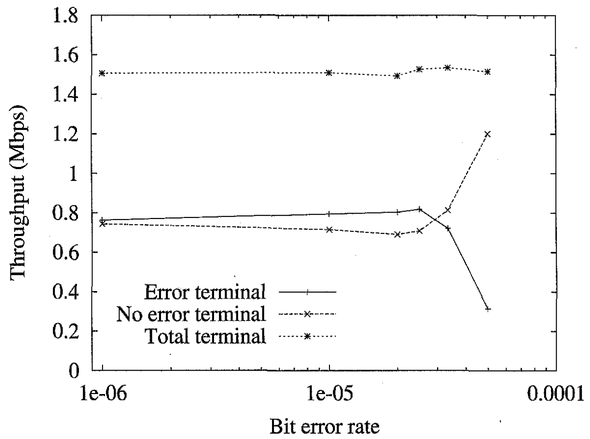


Fig. 20 Average MAC level throughput of Original method when the BERs of two terminals are different (delay 1 ms).

However, the error rate becomes higher, these retransmission controls cannot recover the packet losses and the larger ratio of bandwidth is used by the no error terminal.

3.2.2 Average Queue Length Evaluation

Figure 21 shows the average queue length with the two-

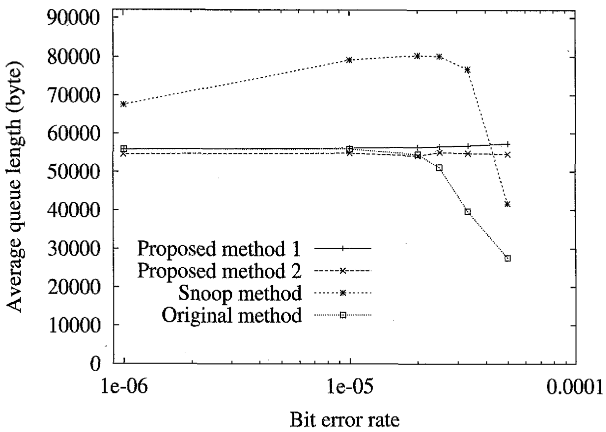


Fig. 21 Average queue length with the two-terminal model (delay 1 ms).

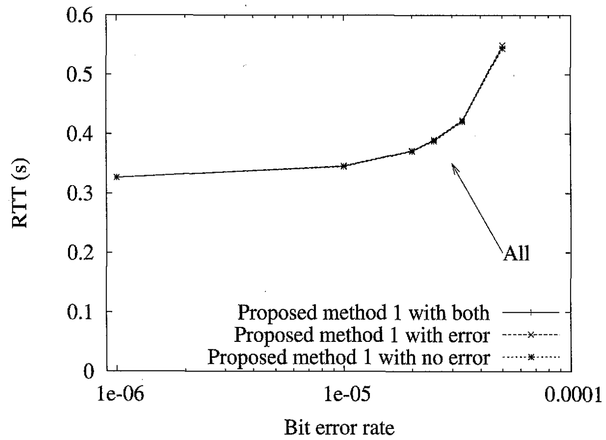


Fig. 22 Average round trip time with two-terminal model in Proposed method 1 (delay 1 ms).

terminal model. In Proposed methods 1 and 2, there is almost no difference under all BER conditions because of the congestion control of two TCP senders and the effect of statistical multiplexing. The original method reduces the queue length for the higher BER condition since the error-prone flow cannot maintain communication and the queue is only needed for the no error terminal. In the snoop method, additional buffer is needed for the snoop cache and this increases the size of queue length. The breaking of TCP semantics also makes the queue length large. In the higher BER conditions, however, the queue length is also reduced for the same reason as with the original method.

3.2.3 Average Round Trip Time Evaluation

Figures 22–25 show the average RTT with the two-terminal model. In Fig. 22, Proposed method 1 shows no difference between the terminals because IEEE 802.11 uses the "stop and wait protocol" and the BS can send just one packet to one of the terminals at a time. Since Proposed method 1 only expands this retransmission control, both flows should wait for almost the same interval in the BS. The increase of the delay with the higher BER congestion is caused by queuing delay in the BS and each terminal.

In Proposed method 2, the bandwidth in MAC layer is

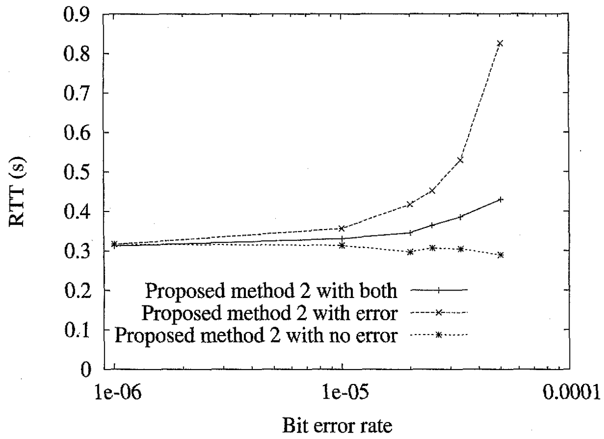


Fig. 23 Average round trip time with two-terminal model in Proposed method 2 (delay 1 ms).

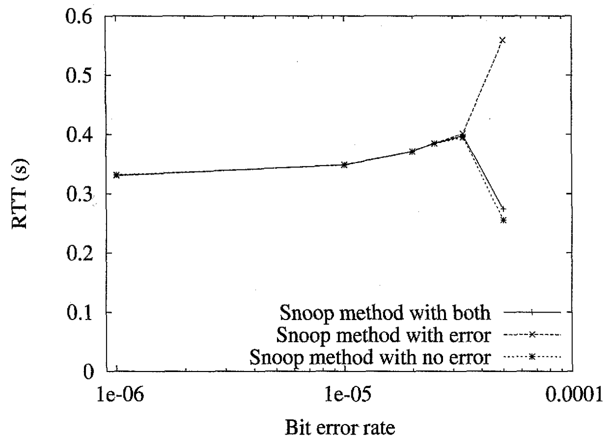


Fig. 24 Average round trip time with two-terminal model in snoop method (delay 1 ms).

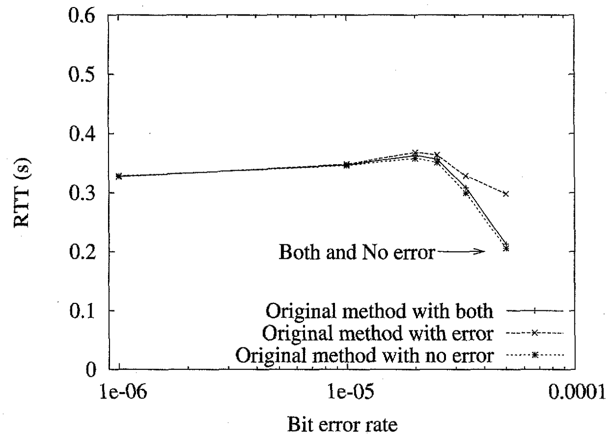


Fig. 25 Average round trip time with two-terminal model in original method (delay 1 ms).

fairly prepared for the two terminals and the average round trip time is almost the same for the no error terminal in every wireless error condition. The heavily retransmission never affects communication of no error terminal. Alternatively, the delay for error prone terminal is increased because packets for error prone terminal have to wait longer period for the retransmissions.

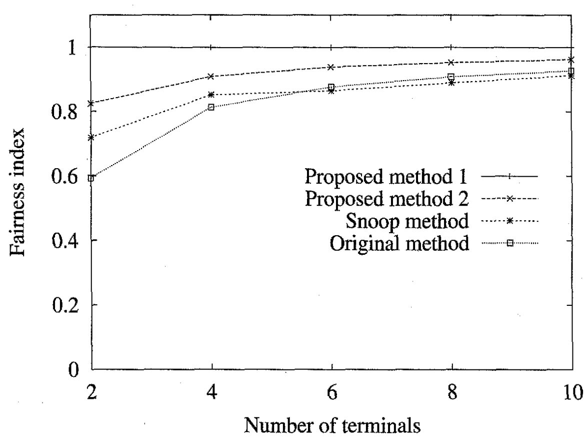


Fig. 26 Fairness index for TCP level goodput when the BERs of terminals are different (delay 1 ms).

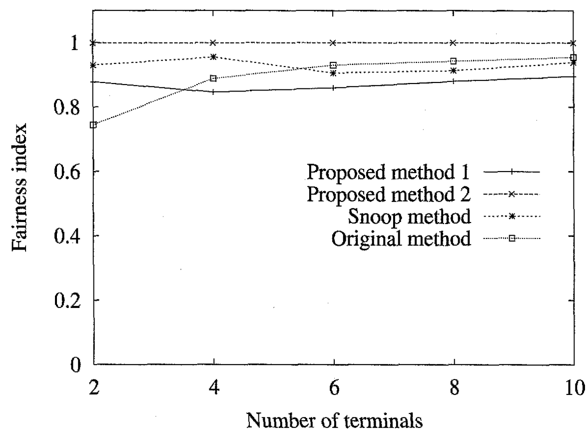


Fig. 27 Fairness index for MAC level throughput when the BERs of terminals are different (delay 1 ms).

The snoop method and the original method have similar tendency for all the conditions in Figs. 24 and 25. In the higher-BER case, the error-prone terminal needs a long RTT because these packets should wait for each local retransmission control. This result also causes low TCP goodput of the error-prone terminal.

3.3 Many Terminals Model (Scalability Evaluation)

This subsection discusses scalability of our proposed methods. We change the number of terminals from 2 to 10 with a wired link delay of 1 ms. The BER of one terminal is fixed at 5×10^{-5} (error terminal), while that of the other terminals are fixed at 1×10^{-6} (no error terminal).

Figure 26 and 27 show a metric of fairness index [18] among the connections for TCP level goodput and MAC level throughput, respectively. Since Proposed method 1 share the bandwidth fairly in IP level, the fairness index is higher than 0.99 for any number of terminals. The fairness index of Proposed method 2 is also higher than those of Snoop method and Original method. These results show that Proposed methods 1 and 2 realize fairly shared bandwidth in TCP level even when there are many terminals with error-prone and no-error links.

In MAC level, Proposed method 2 fairly provides the bandwidth and the fairness index is higher than 0.99 for the number of terminals. Then, the bandwidth is fairly shared with error terminal and no-error terminal in MAC level. To keep fairness in TCP level in heavily error condition, the fairness index of Proposed method 1 becomes lower than that of Snoop method and Original method. However, the fairness index is not changed for the number of terminals and is kept larger than 0.84.

Note that the fairness index approaches 1.0 for the larger number of terminal in the all methods because no-error terminals are assigned the same bandwidth. Then, in the case of that the number of terminals is small, the effect of error-prone terminal becomes large.

Figure 28 and 29 show TCP level goodput of the error terminal and MAC level throughput of the error-terminal, respectively. Figure 28 shows that Proposed methods 1 and 2 keep higher goodput than Snoop and Original methods because the bandwidth for error terminal is kept by the packet scheduling controls. In the case that the number of terminal is larger, Snoop method gives lower goodput than Original method and is difficult to maintain communications. Since the buffer size in BS is limited by 150000 bytes and the

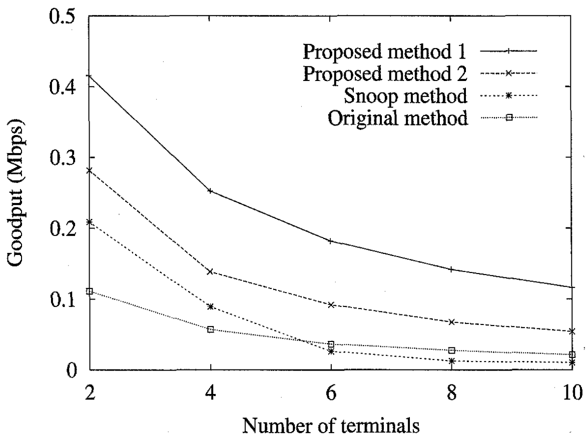


Fig. 28 TCP level goodput for an error terminal when the BERs of terminals are different (delay 1 ms).

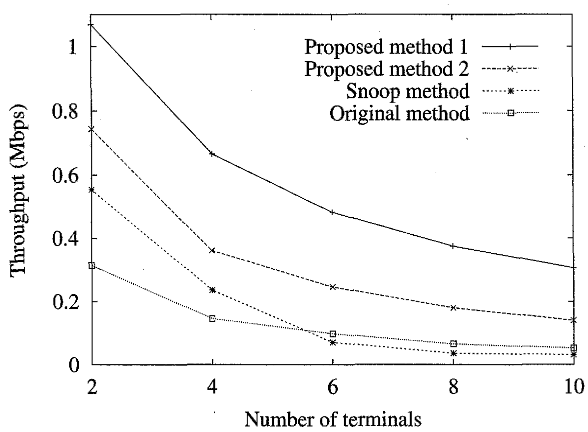


Fig. 29 MAC level throughput for an error terminal when the BERs of terminals are different (delay 1 ms).

snoop cache for each terminal is included in it, packet losses occurs easily with increasing terminals. The influence of limited queue buffer size for the snoop method discussed in section 3.1.4 becomes worse.

In Fig. 29, MAC level bandwidth is fairly shared in Proposed method 2, and this means that the error-prone terminal in Proposed method 1 aggressively obtains MAC level bandwidth to maintain TCP level fairness and the error-prone terminal in Snoop and Original methods is reduced its bandwidth by no-error terminals. Even when the number of no-error terminals becomes large, Proposed method 1 and 2 ensure MAC level bandwidth for the error-prone terminal to maintain the communications and this also realizes scalability of Proposed methods 1 and 2.

4. Conclusion

This paper proposed a cross-layer retransmission control with queue management for TCP communications in wireless environments. Since our proposed methods are designed to reduce the packet retransmission delay and assist MAC local retransmission control, packet losses are completely eliminated from a wireless link and all packets are delivered in the correct order without impact negatively on the conventional TCP congestion controls. From the simulation experiments, the TCP goodput is improved, especially in high BER conditions, and the available bandwidth of the wireless link is used efficiently. The limiting factors come from the limited buffer size in the BS and are also observed. The fairness problem of TCP communication between connections with different bit error rates in a wireless link is also improved, and MAC level fairness is also controllable.

In IEEE 802.11e [19], to serve defferent QoS requirements, separated queue is prepared for them. The different QoS is realized by enhanced distributed channel access (EDCA) or hybrid coordination function controlled channel access (HCCA). However, packet loss in the wireless link in MAC are not considered in these controls. With adding our proposed method to these controls, the communication quality will be improved further.

TCP flows and UDP flows should be dealt separately in our method even if these flows are for one terminal, because that UDP generally delivers time sensitive data. Expanding Proposed method 2, packet delay will be adjustable for time sensitive applications, and a bandwidth will be fairly shared among download and upload traffic. These controls should receive additional consideration in the future.

References

[1] S. Floyd and T. Henderson, "The new reno modification to TCP's Fast Recovery Algorithm," RFC2582, 1999.
[2] M. Mathis, J. Mahdavi, S. Floyd, and A. Romanow, "Selective acknowledgment options," RFC2018, 1996.
[3] A. Chrungoo, V. Gupta, H. Saran, and R. Shorey, "TCP k-SACK: A simple protocol to improve performance over lossy links," Proc. IEEE Globecom'02, pp.1713–1717, 2001.
[4] C. Casetti, M. Gerla, S. Mascolo, M.Y. Sanadidi, and R. Wang,

"TCP westwood: End-to-end congestion control for wired/wireless networks," Wirel. Netw., vol.8, no.5, pp.467–479, 2002.

[5] C.P. Fu and S.C. Liew, "TCP veno: TCP enhancement for transmission over wireless access networks," IEEE J. Sel. Areas Commun., vol.21, no.2, pp.216–228, 2003.

[6] L.S. Brakmo, S.W. O'Malley, and L.L. Peterson, "TCP vegas: New techniques for congestion detection and avoidance," Proc. ACM Sigcom'94, pp.24–35, 1994.

[7] P. Sinha, N. Venkitaraman, R. Sivakumar, and V. Bharghavan, "WTCP: A reliable transport protocol for wireless wide-area networks," Proc. ACM Mobicom'99, pp.231–241, 1999.

[8] D. Barman and I. Matta, "Effectiveness of loss labeling in improving TCP performance in wired/wireless networks," Proc. IEEE ICNP'02, pp.2–11, 2002.

[9] T. Goff, J. Moronski, D.S. Phatak, and V. Gupta, "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments," Proc. IEEE Infocom'00, pp.1537–1545, 2000.

[10] H. Balakrishnan, S. Seshan, and R.H. Katz, "Improving reliable transport and handoff protocols in cellular wireless networks," ACM Wireless Networks, vol.1, pp.469–481, 1996.

[11] H. Balakrishnan and R.H. Katz, "Explicit loss notification and wireless web performance," Proc. IEEE Globecom'98, 1998.

[12] W. Ding and A. Jamalipour, "Delay performance of the new explicit loss notification TCP technique for wireless networks," Proc. IEEE Globecom'01, pp.3483–3487, 2001.

[13] C. Parsa and J.J. Garcia-Luna-Aceves, "Improving TCP performance over wireless network at the link layer," Mobile Networks and Applications, vol.5, pp.57–71, 1999.

[14] M. Shreedhar and G. Varghese, "Efficient fair queuing deficit round robin," Proc. ACM Sigcomm'95, pp.231–242, 1995.

[15] S. Ohzahata, S. Kimura, Y. Ebihara, and K. Kawashima, "A queue management method for improving TCP performance in wireless environments," Proc. IEEE WCNC'04, pp.1069–1075, 2004.

[16] "Network Simulator ns (version2)," http://www.isi.edu/nsnam/ns/

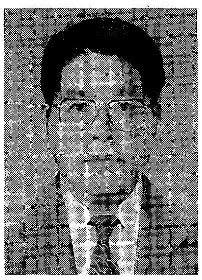
[17] B.P. Crow, I. Widjaja, J.G. Kim, and P.T. Sakai, "IEEE 802.11 wireless local area networks," IEEE Commun. Mag., vol.35, no.9, pp.116–126, 1997.

[18] R.K. Jain, D.W. Chiu, and W.R. Hawe, "A quantitative measure of fairness and discrimination for resource allocation in shared computer systems," DEC Research Report, TR-301, 1984.

[19] "Medium access control (MAC) quality of service enhancements," IEEE Std., 802.11E-2005, 2005.

Satoshi Ohzahata received B.S., M.E., and D.E. degrees from the University of Tsukuba in 1998, 2000 and 2003, respectively. Since 2003 he has been a Research Associate, Department of Computer, Information and Communication Sciences at Tokyo University Agriculture and Technology. He received the Young Engineer Award from the IEICE in 2006. His interests are mobile ad hoc networks, the Internet architecture in mobile environments and Internet traffic measurement. He is a member of IEEE, ACM and IPSJ.

Shigetomo Kimura received the B.E., M.E., and D. Info. Sci. degrees from the Tohoku University, Japan, in 1990, 1992, and 1995, respectively. In 1995, he was appointed an Assistant Professor at Institute of Information Sciences and Electronics, the University of Tsukuba, Japan, becoming an Associate Professor in 2001. Since 2004 he has been an Associate Professor at the Graduate School of Systems and Information Engineering, the University of Tsukuba. His primary research interests are in the areas of algebraic formulation of concurrent processes, and protocols for computer networks. He is a member of ACM, IEEE, IPSJ, and JSSS.



Yoshihiko Ebihara received the B.E., M.E., and D.E. degrees from the Tohoku University, Japan, in 1970, 1972, and 1978, respectively. From 1973 to 1975, he was on the staff of the ALOHA system project at the University of Hawaii. In 1975, he was appointed a Research Associate in the Institute of Electrical Communication, the Tohoku University. He was appointed an Assistant Professor and an Associate Professor at Institute of Information Sciences and Electronics, the University of Tsukuba, in 1976 and 1985, respectively. He was a Professor of the institute from 1993. During 1998–2000, he presided over the Science Information Processing Center of the University of Tsukuba as a director. During 2000–2003, he was a Chair of the Institute. Since 2004 he has been a Professor at the Graduate School of Systems and Information Engineering, the University of Tsukuba. He was a Provost of the Third Cluster of College during 2005–2006, and has been a Provost of the School of Informatics since 2007 in the same university. His primary research interests include computer networks, performance measurement, distributed data base management and digital communication system. He is a member of IPSJ.

Konosuke Kawashima received a B.E. degree from the University of Tokyo, Japan, in 1969, and Dr. Eng. degree from the same university in 1993. He is currently a Professor, Department of Computer, Information and Communication Sciences, Tokyo University of Agriculture and Technology and also Technical Adviser, Director of Teletraffic Research Laboratory of NTT Advanced Technologies Corp (NTT-AT). His research involves teletraffic engineering and science for mobile communications and IP networks. He joined the Electrical Communication Laboratories of NTT in 1969, where he engaged in the research and development of teletraffic engineering for packet networks, mobile communications, and ATM and multimedia networks etc. From 1997 to 2001, he was a consultant for various networks in NTT-AT. He received the Young Engineer Award and Paper Award from the IEICE in 1978, 1982 respectively. He also received the Best Paper Award, Distinguished Achievement and Contributions Award from the Operations Research Society of Japan (ORSJ) in 1986, 2007 respectively, and the Technical Award from the Telecommunications Advancement Foundation in 1996. He is a member of IPSJ, IEEE, ACM/SIGCOMM, IFIP WG6.3, a Fellow of ORSJ and an International Advisory Member for International Teletraffic Congress (ITC).