Performance Improvement of TCP on Wireless Cellular Networks by Adaptive FEC Combined with Explicit Loss Notification

Masahiro Miyoshi†, Masashi Sugano‡, Masayuki Murata†
†Department of Infomatics and Mathematical Science, Graduate School of Engineering Science,
Osaka University 1-3 Machikaneyama, Toyanaka, Osaka 560-8531, Japan
‡Osaka Prefecture College of Health Sciences, 3-7-30 Habikino, Habikino, Osaka 583-8555, Japan

***Abstract*- We propose a new adaptive FEC scheme combined with ELN (Explicit Loss Notification) that was proposed for improving TCP performance in wireless cellular networks. In our method, transmission errors on the wireless link are measured at the packet level and the error status is notified the TCP sender with ELN. According to this information, an appropriate FEC code is determined in order to maximize the TCP performance. We first compare the TCP performance using Snoop Protocol, ELN and the fixed FEC, through which we find the appropriate FEC code against given BER (bit error ratio). We then show how the adaptive FEC can be realized using our solution, and also examine the appropriate observation period of measuring BER enough for the fading speed on the noisy wireless link. We finally demonstrate that our method can achieve better performance than the conventional fixed FEC by using the Gilbert model as a wireless error model.**

I. Introduction

A mobile Internet technology based on a wireless cellular network has been rapidly deployed in these several years. In the Internet, TCP is used as a standard transport layer protocol. TCP recognizes a congestion occurrence within the network by a packet loss, and performs congestion control by throttling the congestion window. Then, we encounter the problem that the packet losses due to the transmission errors cause unexpected degradation of TCP throughput in a wireless cellular network environment. Many approaches have been proposed to improve TCP throughput in the wireless cellular network. The split connection approach such as Indirect-TCP [1] is an early proposal in the field. It involves splitting each TCP connections between a sender and a receiver into two separate connections. Snoop Protocol [2] is an improved scheme of split connection approaches, and it retains end-to-end semantics; it uses Snoop Agent to cache the TCP segment and retransmits the segment only on the wireless link. ELN (Explicit Loss Notification) [3] is a more precise approach with a capability of distinguishing the packet loss due to congestion from the one by transmission errors. Another approach to improving the TCP performance is to enhance the lower layer protocol, i.e., the data link layer protocol, by incorporating, e.g., ARQ (automatic repeat request) and/or FEC (forward error correction) [4]. FEC is a simplest solution to improve the bit error ratio seen by the higher layer protocol. However, it is inefficient if an error condition of the radio channel varies greatly. Accordingly, an adaptive error correction scheme is proposed in [5] in order to compensate such a drawback of FEC. However, utilizing the adaptive error correction solely is not a realistic solution because it is complicated to measure BER (bit error rate) for each bit or the unit of several bits and change the FEC code appropriately against irregular wireless errors. In reference [5], it is shown an adaptive error correction method for wireless LAN where the packet size and the degree of FEC redundancy are adaptively controlled according to the packet error rate. But it is needed specially-formatted UDP datagrams as probed packets to measure the packet error rate, and that these datagrams compress available bandwidth. In short, it has not been proposed Adaptive FEC scheme using no probed packets. In this paper, we propose a new adaptive FEC scheme combined with ELN, and we show a method for deciding threshold value to change adaptive FEC code, which does not clarified by [5]. In our method, transmission errors on the radio link are measured at the packet level and the error status is notified the TCP sender with ELN. According to this information, an appropriate FEC code is selected. We first evaluate the TCP performance using Snoop Protocol, ELN and the fixed FEC, through which we find the appropriate FEC code against given BER. We will then show how the adaptive FEC can be realized using our solution, and also examine the appropriate observation period of measuring BER enough for the fading speed on noisy wireless link. We will finally evaluate our proposed method by Gilbert model [6] as wireless error model, and show that our method achieves better performance than the conventional fixed FEC. The rest of this paper is organized as follows. In Section II, we show the network model in our study. In Section III, we evaluate the TCP performance using Snoop Protocol, ELN and the fixed FEC. We describe our proposed method in Section IV, and evaluate it in Section V. We finally describe concluding remarks in Section VI.

II. Network Model

We use the network configuration shown in Figure 1 with IMT-2000 support [7] to investigate the TCP performance. Parameters are summarized in Table 1. Here, the bandwidth of the downlink is 384kbps for best effort service in the first deployment in Japan. In this configuration, since the major application of the current Internet is to download the Web documents from the fixed server to the wireless terminal, we assume that TCP segments are transmitted towards the wireless terminal from the wired terminal, and ACK segments are in the reverse direction. We consider that packet loss is mainly caused by the transmission error on the wireless link, and the packet loss due to buffer overflow is assumed to be negligible.

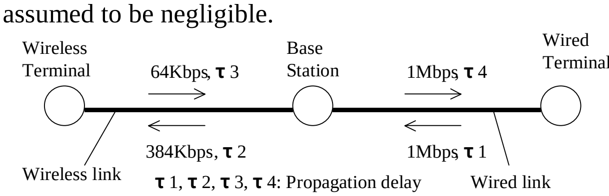


Figure 1 Network model

Table 1 Network parameters

TCP segment size	100 byte
ACK size	40 byte
Buffer size	50 kbyte
Propagation delay (τ 1, τ 2, τ 3, τ 4)	1ms

III. Performance Comparison for TCP on Wireless Cellular Network

In this section, we first describe major schemes of improving for TCP throughput on wireless cellular network, and then evaluate TCP throughput for major schemes.

TCP provides reliable end-to-end data communication using the following two main congestion control mechanisms. The one is Fast Retransmit and Fast Recovery mechanisms, which throttle the congestion window size to half, if the TCP sender detects triple duplicate ACKs continuously. The other is a retransmission timeout mechanism, which draws back the congestion window size to 1 MSS (Maximum Segment Size), if the ACK for the TCP segment is not received before retransmission timeout timer expiration.

Those congestion control mechanisms perform well on wired links since the most of packet losses occurs due to congestion. However, on wireless links, it does not because the packet loss may occur due to transmission errors and TCP cannot distinguish the packet loss due to congestion and transmission errors. Therefore, it is well known that the packet losses due to transmission errors cause unexpected degradation of TCP throughput in a wireless cellular network environment.

Many approaches have been proposed to improve TCP throughput in the wireless cellular network. These are summarized in the IETF (Internet Engineering Task Force) [8]. Here, we describe Snoop Protocol [2], ELN [3], and the link-layer scheme as major schemes.

- Snoop Protocol

Snoop Protocol is an improved scheme of split connection approaches. It retains end-to-end semantics of TCP, such that it uses Snoop Agent to cache the TCP segment and retransmits the segment only on the wireless link. Snoop Protocol performs the following sequence in the case of the wired terminal being the TCP sender.

(1) The TCP segments are cached at the base station.

(2) The TCP segments are retransmitted at the base station, if packet loss is detected on the wireless link.

- ELN

ELN is an approach that enables distinguishing the packet loss due to congestion from transmission errors. If the packet loss occurs due to transmission errors, the TCP sender takes appropriate actions. That is, ELN performs the following sequence in the case of the wired terminal being the TCP sender.

(1) The TCP sequence numbers are cached at the base station.

(2) The ACK packet is attached "ELN bit active" at the base station, if the packet, of which sequence number is cached, is lost.

(3) The TCP sender takes an appropriate action when receiving "ELN bit active". One typical realization is that the TCP sender does not perform congestion control.

- The Data Link layer scheme

Enhancements of the link-layer protocol include ARQ (automatic repeat request) and FEC (forward error correction) [4]. FEC is a simplest solution to improve the bit error ratio seen by the higher layer protocol. However, it is inefficient if an error condition of the radio channel varies greatly. Accordingly, an adaptive error correction scheme is proposed in [5] in order to compensate such a drawback of FEC. However, utilizing the adaptive error correction solely is not a realistic solution because it is complicated to measure BER (bit error rate) for each bit or the unit of several bits and change the FEC code appropriately against irregular wireless errors.

We evaluate TCP throughput by using the TCP Reno version, which is a current de facto standard for TCP implementations. For enhancement of the TCP performance, we consider the following three schemes.

- ELN, which distinguishes between packet loss due to congestion and others.
- Snoop+ELN, which is a combination of Snoop and ELN.
- FEC as an enhancement of the data link layer protocol.

We used the ns-2 simulator [9]. Simulation results are shown in Figure 2, where the horizontal axis shows the bit error rate on the wireless link. We adopt BCH (127, 92) as the FEC code. It can be observed in the figure that FEC is most effective to prevent TCP throughput degradation in the range where BER is greater than 0.0001. On the other hand, in the lower BER region, TCP throughput of FEC becomes smaller than other schemes due to its overhead and Snoop+ELN achieves the best performance.

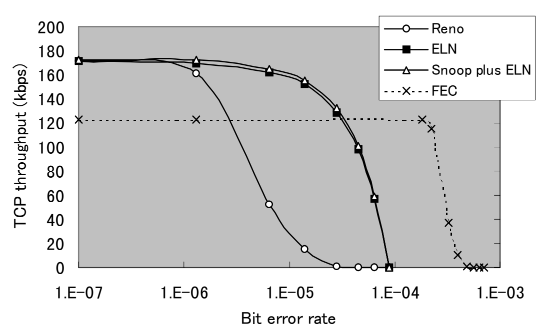


Figure 2 Comparison of TCP throughput

IV. Adaptive FEC Combined with Snoop+ELN

In Figure 3, we present the TCP throughput of Snoop+ELN combined with FEC. The FEC code is changed as BCH (127, 92) as 5-bit correct code, BCH (127, 64) as 10-bit correct code and BCH (127, 36) as 15-bit correct code. The figure clearly shows an existence of the optimal FEC parameter to achieve the best performance dependent on BER. In short, an adaptive FEC combined with Snoop+ELN scheme is more effective in order to prevent TCP throughput degradation. In this paper, we propose a new adaptive FEC scheme combined with ELN, and we show a method for deciding threshold value to change adaptive FEC code, which does not clarified by [5].

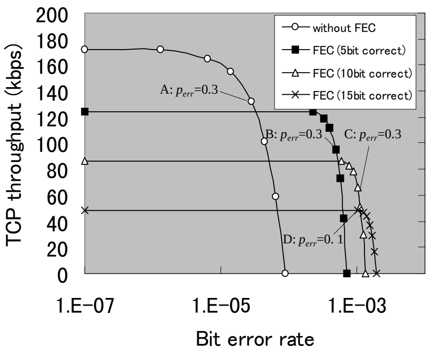
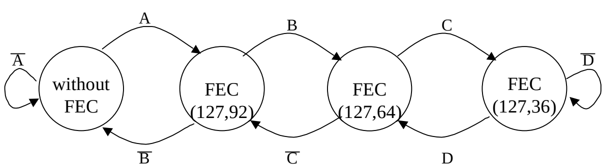


Figure 3 TCP throughput on various FEC code combined with Snoop+ELN



A: p_{err}>0.3 (In the case of two or more packets in a window of ten)
B: p_{err} > 0.3 (In the case of two or more packets in a window of ten)
C: p_{err} > 0.3 (In the case of two or more packets in a window of ten)
D: p_{err}<0.1 (In the case of two or more packets in a window of ten)

Figure 4 Method of changing adaptive FEC code

In the current Internet, network congestion has been solved to improve network infrastructures. We consider that packet loss is mainly caused by the transmission error on the wireless link, and the packet loss due to network congestion is assumed to be negligible. In this assumption, TCP senders recognize the transmission error on the wireless link by ELN, and perform no congestion control.

Of course, we should take account of the fact that the packet error rate is dynamically changed according to the condition of the wireless link. In the adaptive FEC scheme, it is important how FEC code should be changed according to the packet error rate. In this paper, we adopt a reactive-based adaptive algorism by observed errors. In the reactive-based adaptive algorism, it is important that the algorism is robust in the sense that it is not influenced by the sensitive errors. The authors in [5] proposed a technique of adaptively changing FEC; the FEC code is changed when two or more segments among ten segments encounter the transmission error. Our proposed method is based on that idea. Based on the above numerical results, we show a method for changing the FEC capability according to p_{err} obtained by ELN as shown in Figure 4. Here, the current FEC code is changed to the more error correctable code when p_{err} exceeds the threshold value. Otherwise, it is changed to the less overhead code. In Figure 3, the symbol of the A, B, C, and D is the threshold value of p_{err} and correspond to the same symbol in Figure 4.

We finally examine the time period of measuring p_{err}. The smaller value of the time period is expected so that it can be adapted to the time changing packet loss rate. However, it must be large enough for the fading speed. The fading speed is given by f_d (Doppler frequency) [10] as:

$$f_d = \frac{v\ f_c}{c} \tag{1}$$

where f_c is the carrier frequency, v is the vehicle speed, and c is the speed of light. For the 900-MHz carrier frequency, the above formula yields a Doppler frequency of 3Hz/m/sec. Therefore, for the high-speed wireless transmission (e.g., in the order of Mbps), the fading speed can be viewed as a roughly constant value for the time period of measuring p_{err} (e.g., 10 sampling with the frame length of 1000-byte segment).

V. Evaluation of Adaptive FEC Combined with Snoop+ELN

In this section, we evaluate our proposed method by the Gilbert model [6] as a wireless error model. It has been widely used to model the noisy link with time-varying errors. In the Gilbert model, two states of "Error-free" and "Error" are expressed in terms of transition probabilities p_{GE} and p_{EG}, and average error-free length L_G and error length L_E. See Figure 5.

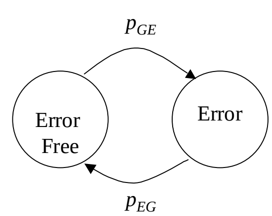


Figure 5 Gilbert model

$$p_{GE} = \frac{1}{L_G} \tag{2}$$

$$p_{EG} = \frac{1}{L_E} \tag{3}$$

The Gilbert model is a two-stare Markov model and each state is memoryless. Recalling that the geometric distribution is a discrete equivalent of the exponential distribution [11], we determine the length $G(p)$ of staying in each state as follows.

$$G(p) = \frac{\ln(u)}{\ln(1-p)} \tag{4}$$

where u is a random number uniformly distributed from 0 to 1, and p is the leaving probability from the state.

Using the above model, we generate wireless errors and simulate whether lost or success for each sending packet. See the flow chart illustrated in Figure 6.

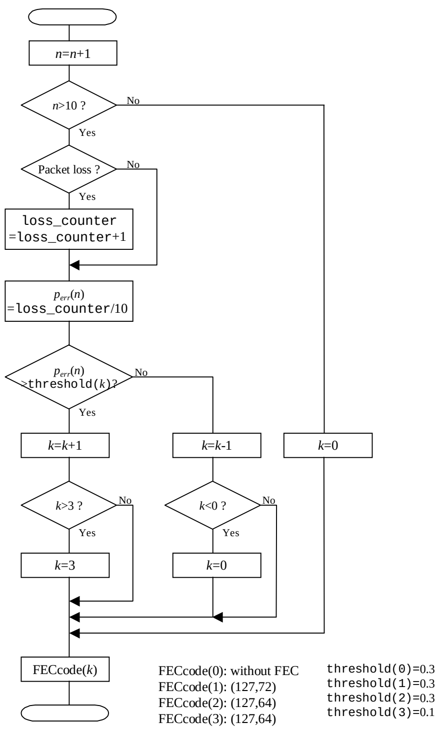


Figure 6 Flow of Adaptive FEC Combined with Snoop+ELN

In simulation, the number of sending packets is 50,000, and packets length is fixed at 800 bits, which is commonly seen in the mobile phone data transfer services. The parameter set of the Gilbert model is shown in Table 2.

Table 2 Gilbert model parameters

	p_{GE}	p_{EG}
lower BER case	0.006	0.05
higer BER case	0.006	0.007

Here, we assume that bit errors in each packet occur randomly. We show the simulation results in Table 3 in the case of a rather lower value of BER (p_b = 0.00015, p_{err} = 0.090), and in Table 4 in the case of the higher BER (p_b = 0.00075, p_{err} = 0.445), respectively. We also show the values obtained through the simple analysis for the case without FEC and for the one with fixed FEC. See appendix for the analyses. In the table, the second column shows the number of lost packets by wireless packet errors and the third column does the number of received packets. The fourth column shows the number of overhead packets, which is the case with fixed FEC obtained by Eq. (a-5). The final column shows the number of successfully received packets, in which we omit the number of overhead packets from the number of totally received packets. We can observe that our adaptive FEC can achieve the best performance. That is, by using the adaptive FEC, the successfully received packets are improved by 2% in Table 3, and by 10% in Table 4. It is because our adaptive FEC encounters less packet losses than without FEC, and needs less overhead packets than the fixed FEC.

Table 3 Comparison of received packets
in the case of lower BER (pb = 0.00015, perr = 0.090)

	lost packets	received packets	over head packets	received packets (available)
without FEC	4461	45539	0	45539
FEC(127,94)	35	49965	12983	36982
FEC(127,64)	0	50000	24803	25197
FEC(127,36)	0	50000	35827	14173
Adptive FEC	2292	47708	1002	46706

Table 4 Comparison of received packets
in the case of higer BER (pb = 0.00075, perr = 0.445)

	lost packets	received packets	over head packets	received packets (available)
without FEC	22242	27758	0	27758
FEC(127,94)	15903	34097	8860	25237
FEC(127,64)	934	49066	24340	24726
FEC(127,36)	11	49989	35819	14170
Adptive FEC	8213	41787	9223	32564

VI. Concluding Remarks

In this paper, we have proposed a new adaptive FEC scheme combined with ELN. In our method, transmission errors on the radio link are measured at the packet level and the error status is notified the TCP sender with ELN. According to this information, an appropriate FEC code is selected. We have evaluated the TCP performance using Snoop Protocol, ELN and the fixed FEC, through which we found the appropriate FEC code against given BER. We have also shown how the adaptive FEC can be realized using our solution. We have finally evaluated our proposed method by the Gilbert model as the wireless error model, and show that our method achieves better performance than conventional fixed FEC.

ACKNOWLEDGMENT

This work was partly supported by Research for the Future Program of Japan Society for the Promotion of Science under the Project "Integrated Network Architecture for Advanced Multimedia Application Systems," Special Coordination Funds for promoting Science and Technology of the Science and Technology Agency of the Japanese Government, Telecommunication Advancement Organization of Japan under the Project "Global Experimental Networks for Information Society Project," a Grant-in-Aid for Scientific Research (A) (2) 11305030 from The Ministry of Education, Science, Sports and Culture of Japan.

APPENDIX

In this appendix, we show how the numbers of overhead packets and successfully received packets are calculated in the case of without FEC and fixed FEC.

[Without FEC]

By assuming that the bit error in each packet occurs randomly, packet error rate p_{err} is represented by using the bit error rate p_b and the packet length l as:

$$p_{err} = 1 - (1 - p_b)^l \qquad \text{(a-1)}$$

Using p_{err}, the number of lost packets, P_{loss}, is determined as:

$$P_{loss} = p_{err} P_{send} \qquad \text{(a-2)}$$

Similarly, the number of successfully received packets, P_{rcv}, is given by:

$$P_{rcv} = P_{send} - P_{loss} \qquad \text{(a-3)}$$

[Fixed FEC]

If the bit errors in each packet occur randomly, p_{err} is represented by the following binomial distributed expression by using bit error rate p_b, packet length l and error correct ability c.

$$p_{err}(l, c) = 1 - \sum_{i=0}^{c} \binom{l}{i} p_b^i (1 - p_b)^{l-i} \qquad \text{(a-4)}$$

Using packet length l and data length d, overhead by fixed FEC is calculated by following expression:

$$H = \frac{d}{l} \qquad \text{(a-5)}$$

Using p_{err} and overhead H, the numbers of lost packets P_{loss}, overhead packets P_{head} , and successfully received packets P_{rcv} are obtained by the following expressions:

$$P_{loss} = p_{err} P_{send} \qquad \text{(a-6)}$$

$$P_{head} = HP_{send} \qquad \text{(a-7)}$$

$$P_{rcv} = P_{send} - P_{loss} - P_{head} \qquad \text{(a-8)}$$

REFERENCES

[1] A. Bakre and B. R. Badrinath, "I-TCP: Indirect TCP for mobile hosts," *Proc. of IEEE International Conference on Distributed Computing Systems* (*ICDCS '95*), pp.136-143, May 1995.

[2] H. Balakrishnan, S. Seshan and R. H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," *Wireless Networks*, vol.1, Dec. 1995.

[3] H. Balakrishnan and R. H. Katz, "Explicit loss notification and wireless web performance," *Proc. IEEE GLOBECOM '98 Internet Mini-Conference,* Nov. 1998.

[4] E. Ayanoglu, S. Paul, T. F. LaPorta, K. K. Sabnani, and R. D. Gitlin. "AIRMAIL: A link-layer protocol for wireless networks," *Wireless Networks*, vol.1, pp.47-60, Feb. 1995.

[5] D. A. Eckhardt and P. Steenkiste, "A trace-based evaluation of adaptive error correction for a wireless local area network," *Mobile Networks and Applications*, vol.4, no.4, pp. 273-287, Dec. 1999.

[6] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Systems Technical Journal*, vol. 39, pp. 1253-1265, Sept. 1960.

[7] Available at http://www.itu.int/imt/.

[8] G. Montenegro, S. Dawkins, M. Kojo, V. Magret and N. Vaidya, "Long Thin Networks," *RFC2757*, Jan. 2000.

[9] Available at http://www.isi.edu/nsnam/ns/.

[10] A. Kumar and J. Holtzman, "Comparative performance analysis of versions of TCP in a local network with a Mobile Radio Link," *IEEE/ACM Transactions on Networking*, vol. 6, no. 4, pp. 485-498, Aug. 1998.

[11] R. Jain, The Art of Computer Systems Performance Analysis: Techniques for Experimental Design, Measurement, Simulation, and Modeling, *Wiley*, 1991.