

# Transmission Control for Future Internet including Error-prone Wireless Region

I. Ryoo, S. Kim

**Intae Ryoo, Seokhoon Kim**

Dept. of Computer Engineering & RU-IPTV Research Center  
Kyunghee Univ., 1, Seocheon-dong, Giheung-gu,  
Yongin-si, Gyeonggi-do, Korea  
E-mail: itryoo@khu.ac.kr, kimsh@khu.ac.kr

## **Abstract:**

This paper introduces a transmission control scheme which is aimed at enhancing overall transmission capability of internetworks including *error-prone wireless regions*. The proposed scheme can accurately adapt to wireless communication environments by integrating new approaches of *bandwidth estimation*, *loss detection*, and *error recovery* techniques while differentiating data losses due to physical bit error characteristics of *error-prone wireless regions* from those due to network congestion. From the simulations, it has been verified that the proposed scheme shows better throughput performances than the existing major transmission control schemes such as New Reno, Vegas, and Westwood+, while achieving satisfied levels of fairness and friendliness with them.

**Keywords:** transmission control, error-prone wireless region, bandwidth estimation, loss detection, error recovery

## 1 Introduction

Wireless and mobile communication environments have already been pervasively deployed, and never-ending communication demand in error-prone wireless regions of upcoming Future Internet makes it difficult to maintain the same level of performances of legacy transmission controls. This is mainly due to the inability of the existing transmission control schemes to differentiate data losses resulting from inevitable bit errors in error-prone wireless regions [1] from those resulting from network congestion. As bad channel conditions and aperiodic disconnections are typically transient phenomena, legacy TCP congestion control responses may be inappropriate and undesirable for the networks with error-prone wireless regions. In order to achieve a higher level of transmission efficiency, we have proposed a new transmission control method that is well suited for error-prone wireless regions as well as wired regions.

Many solutions have been proposed to improve TCP performances in wireless networks and can be categorized into End to End (E2E), Split Connection, and Link Layer solutions [2][3]. The proposed transmission control protocol for error-prone wireless regions (TCP EWR) falls into the E2E category as it measures packet arrival rates and acts according to current network status. It, however, differs from the general E2E solutions in the point that it is aiming to improve its performances in case of random or sporadic data losses by performing a time-stamp based bandwidth estimation at the receiving TCP rather than at the sending TCP, calculating the expected and actual bandwidths, and reacting accordingly to the random wireless packet loss events. It measures bandwidth utilization by using packet arrival information at the receiver rather than by using acknowledgement (*ACK*) information at the sender like TCP New Jersey [4]. As a result, sending feedback information to the receiver is not necessary in our scheme. Moreover, in order to make the TCP EWR operate in a stable status, we have introduced two thresholds,  $\alpha$  and  $\beta$ , which corresponds to having too little and too much data enroute to the destination, respectively. In addition, the TCP EWR performs an enhanced error recovery in

case that the network suffers from timeout expiration due to wireless packet losses. That is, if there is any sign of data loss either by retransmission timeout or by three duplicate *ACK*s, the TCP EWR sets slow start threshold (*ssthresh*) value to optimal congestion window (*oCwnd*). With these features, the overall performance of the TCP EWR can be increased for the networks including error-prone wireless regions.

In Sect. 2, we discuss the bandwidth estimation, loss detection, enhanced error recovery, and congestion control procedures of the TCP EWR. We also give a pitch of our scheme through performance tests in Sect. 3, and finally bring this paper to a conclusion in Sect. 4.

## 2 Transmission Control for Future Internet Including Error-Prone Wireless Regions

The proposed transmission control scheme is composed of bandwidth estimation at the receiver, loss detection, enhanced error recovery, and congestion control. The first step of bandwidth estimation is to calculate a sample bandwidth ( $Bw_{sample}(n)$ ) when the  $n^{th}$  packet arrives at the receiver at time  $t_n$ :

$$Bw_{sample}(n) = d_n / (t_n - t_{n-1}) \quad (1)$$

where  $d_n$  is the amount of data currently received by the receiver and  $t_{n-1}$  is the previous  $(n-1)^{th}$  packet arrival time. That is, differently from the existing schemes such as Westwood+ and New Jersey [5], our scheme uses packet arrival information at the receiver instead of using returning *ACK* information at the sender. Note that Westwood+ and New Jersey are considered in this work as they are representative transmission control schemes that can distinguish wireless packet losses from congestion packet losses and react accordingly. By incorporating the actual arrival rate information in bandwidth estimation and, at the same time, by including congestion-related packet loss probability into the packet arrival rate calculation [6], the proposed scheme can probe the available bandwidth more accurately. The next step is to estimate the available bandwidth with a time varying coefficient, exponentially weighted moving average filter, which yields to following equation.

$$Bw_{estimated}(n) = Bw_{estimated}(n-1) \times \delta_n + Bw_{sample}(n) \times (1 - \delta_n) \quad (2)$$

where  $\delta_n$  is a constant filter gain. In our simulation, we set  $\delta_n$  to 0.7 for  $n \geq 2$  and zero for  $n=1$  for simplicity. Thus, the optimal congestion window size  $oCwnd_n$  is computed as following:

$$oCwnd_n = RTT_{min} \times Bw_{estimated}(n) / seg\_size \quad (3)$$

where  $RTT_{min}$  is minimum measured round-trip time (*RTT*) and *seg\_size* is the length of the TCP segment. Note that we do not consider the available buffer space in the receiver in this work, and  $oCwnd_n$  is used as a receiver-advertised congestion window size. Simulation results in Sect. 3 show that this yields good bandwidth estimates for random packet loss scenarios with error-prone wireless region.

In order to proactively detect the incipient stage of congestion and to efficiently figure out the reason for packet losses, the proposed scheme calculates maximum expected throughput and measures actual throughput as followings similar to Vega and Veno approaches [7]:

$$expected\ throughput = WindowSize / RTT_{min} \quad (4)$$

$$actual\ throughput = WindowSize / RTT \quad (5)$$

where  $RTT$  is the smoothed round-trip time measured. The difference  $D$  between these two throughputs indicates the amount of data that is currently residing on the corresponding path and/or going away due to network congestion or error-prone wireless regions. Also, by using the backlog variable  $x = D * RTT_{min}$  and two thresholds,  $\alpha$  and  $\beta$ , which correspond to having relatively a few and too many data on the route respectively, we can grasp how to manage the sender's congestion window ( $cwnd$ ) size based on the current network situation. In our simulations, the values of  $\alpha=1$  and  $\beta=9$  are used, but they can be set to better adapt to Future Internet environment, which is for further study. With these indicators for the current network status, the TCP EWR performs an enhanced error recovery (*EER*) mechanism. If there is any sign of data loss either by retransmission timeout (*RTO*) at the sender or by three duplicate *ACKs* (*DUPACKs*) from the receiver, the proposed scheme sets the *ssthresh* value to the  $oCwnd_n$ . Note that, if the connection is restored after physical bit errors, the window size should grow quickly to make up for the previous data loss event. In the proposed scheme, the sender immediately adjusts its congestion window size to the  $oCwnd_n$ . It is because the receiver has continuously monitored the available bandwidth of the corresponding path as given in Eq. (1) and (2). There is no need to employ any kinds of slow start, additive increase, or fast recovery algorithms whenever the connection is recovered.

With regard to congestion window updates when a packet loss is detected, there are three different cases depending on the values of backlog variable  $x$ , and two thresholds  $\alpha$  and  $\beta$ . When  $x < \alpha$ , the proposed scheme updates the congestion window size  $cwnd$  to  $oCwnd_n$  or follows an additive-increase paradigm (linear increase state) by comparing the current  $cwnd$  with a new *ssthresh* value. The reason why the proposed scheme updates the  $cwnd$  like this is that the network condition is not too bad as the backlog  $x < \alpha$  implies, although there exists an incident that packets have been lost during transmission. When  $\alpha \leq x \leq \beta$ , it updates the congestion window size to  $oCwnd_n$  or keeps the previous  $cwnd$  value unchanged by comparing the current  $cwnd$  with a new *ssthresh* value. The backlog  $\alpha \leq x \leq \beta$  implies that there is a high probability that packets will be lost due to either network congestion or wireless bit errors. So, it is desirable to retain the previous  $cwnd$  (wait-and-see state) rather than following an additive-increase/adaptive-decrease paradigm. When  $x > \beta$ , it decreases the congestion window size adaptively. The backlog  $x > \beta$  means that the network falls into a congestion state. By adopting this EER approach, we can use the available bandwidth efficiently. The pseudo code of the proposed TCP EWR transmission control algorithm is given below:

- a) When packets successfully arrive at the receiver:  
 $Bw_{sample}(n)$ ,  $Bw_{estimated}(n)$ , and  $oCwnd_n$  are computed;
- b) On ACK reception at the sender:  
 $cwnd = oCwnd_n$ ;
- c) When 3 *DUPACKs* are received or *RTO* expires:  
 if ( $x < \alpha$ )  
 $ssthresh = oCwnd_n$ ;  
 if ( $cwnd > ssthresh$ )  
 $cwnd = oCwnd_n$ ;  
 else /\* linear increase state \*/  
 $cwnd = cwnd + 2$ ;  
 else if ( $\alpha \leq x \leq \beta$ )  
 $ssthresh = oCwnd_n$ ;  
 if ( $cwnd > ssthresh$ )  
 $cwnd = oCwnd_n$ ;  
 else /\* wait-and-see state \*/  
 $cwnd$  is retained;

```

end if
else /* congestion state */
  ssthresh = oCwndn;
  cwnd = 1;

```

Note that the *cwnd* is increased by two in the linear increase state because it is updated after one *RTO* rather than after each *RTT*.

### 3 Simulation Results

The proposed transmission control scheme has been simulated by using NS-2 and its performances are compared with several representative schemes such as New Reno, Vegas, Westwood+, and New Jersey. Performance metrics used in simulations include throughput, fairness, and friendliness. Although the TCP EWR is proposed for adapting to Future Internet including error-prone wireless regions, it must also operate well for any topology with lossy or congested links. Figure 1 shows a single bottleneck scenario for comparing performance metrics of the TCP EWR with those of well-known schemes under the condition that there exist various link error rates, and background traffics.

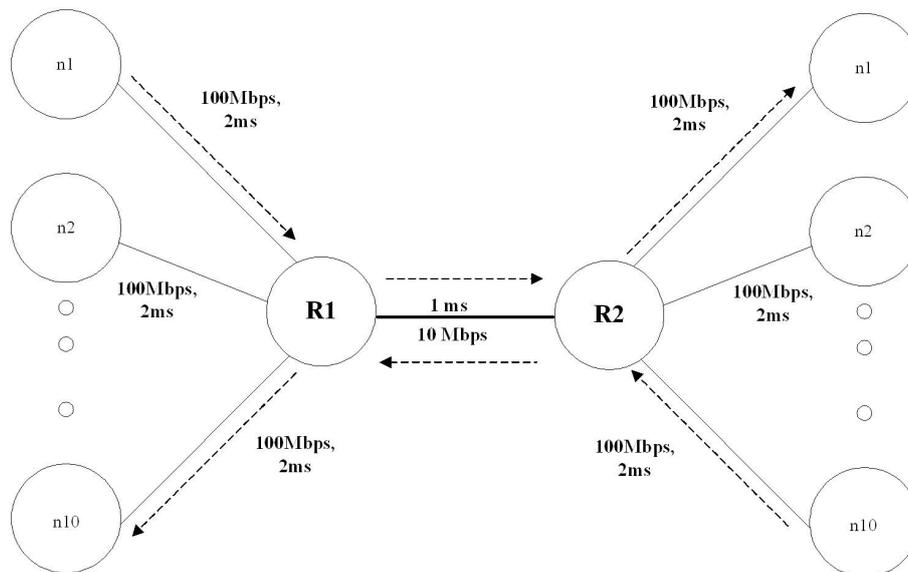


Figure 1: Single Bottleneck Scenario

Figure 2 shows throughput comparison results. The TCP EWR shows almost similar performances when link error rate is low. However, as the link error rate increases, especially from the point of 8 % link error rate, the TCP EWR outperforms other schemes. In addition, fairness index [8] has been compared by considering 10 same TCP flows that share a 10 Mbps bottleneck link. Different TCP schemes have been simulated individually and the corresponding results are summarized in Table 1. A perfect fairness of bandwidth allocation leads to the fair index of 1. It has been verified that, except 0 % link error rate, the TCP EWR's fairness index achieves a satisfactory margin compared to other schemes. The reason why the TCP EWR shows an inferior result under the very low link error condition is that it does not aggressively use the bandwidth but maintain the optimal congestion window size  $oCwnd_n$ .

Figure 3 shows a mixed wired and error-prone wireless scenario where there exist multiple

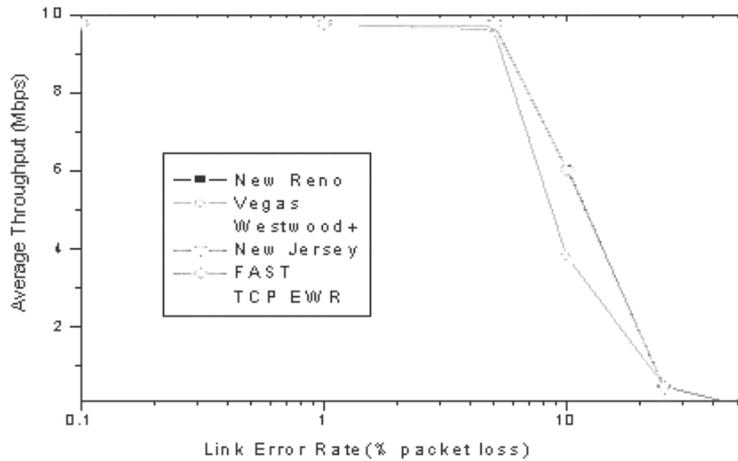


Figure 2: Throughput Comparison for Single Bottleneck Scenario

Table 1: Fairness Comparison for Single Bottleneck Scenario

Error rate(%)	New Reno	Westwood+	EWR
0	0.76	0.80	0.64
0.1	0.31	0.93	0.98
1	0.25	0.74	0.76
10	0.57	0.23	0.62

TCP connections. In this scenario, bidirectional FTP background traffics flow between N1 and N3 and between N2 and N4. The queue sizes of wired links and the wireless link are set to 100 and 10, respectively.

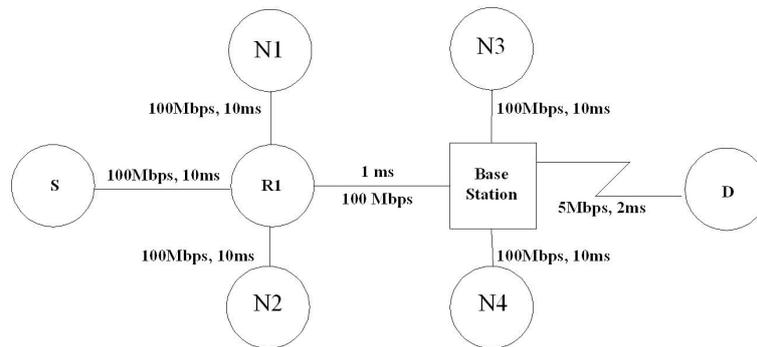


Figure 3: Mixed Wired/Error-prone Wireless Scenario

With these simulation conditions, we have compared the TCP EWR with New Reno, Vegas, and Westwood+. New Reno is considered as it is the leading Internet congestion control scheme. Vegas is considered as it also proposes, as Westwood+, a new mechanism to throttle the congestion window based on measuring the network congestion status. Westwood+ is considered as it remarkably improves utilization of wireless links that are affected by losses not due to congestion. Figure 4 shows the corresponding throughput comparison results. New Reno, Westwood+, and EWR show almost similar throughput results when the link error rate is less than 1 %. When

the link error rate is greater than 1% and less than 10 %, the EWR outperforms Westwood+. Vegas shows poor performance throughout the whole link error range. From the results shown in Figure 2 and Figure 4, we can conclude that the EWR shows better throughput performances than the other schemes for wireless scenario as well as for wired scenario. The reason why the EWR shows again similar result as New Reno and Westwood+ is that the link with more than 10 % error rate makes all the above control schemes fall into congestion state and cannot be logically managed any more.

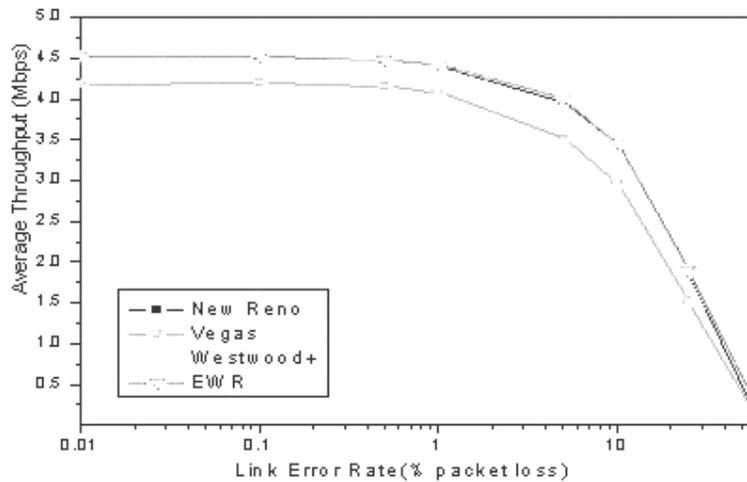


Figure 4: Throughput Comparison for Mixed Wired/Error-prone Wireless Scenario

For the same topology shown in Figure 3, fairness indices of New Reno, Westwood+, and the EWR are compared. There are seven TCP flows of the same version that share a bottleneck link of which the bandwidth is 5 Mbps. The results are shown in Table 2. From these results, we can verify that the EWR also achieves a satisfactory level of fairness index as New Reno and Westwood+ do.

Table 2: Fairness Comparison for Mixed Wired/Error-prone Wireless Scenario

Error rate(%)	New Reno	Westwood+	EWR
0	0.95	0.94	0.96
0.1	0.94	0.94	0.89
1	0.95	1.00	0.95
10	0.96	0.95	0.96

Table 3: Friendliness Comparison of the EWR with New Reno (Throughputs in Kbps)

Number of Connections(New Reno)	Number of Connections(EWR)	Mean Throughput (New Reno)	Mean Throughput (EWR)
3	7	69.24	59.97
5	5	77.91	77.68
7	3	71.55	71.03

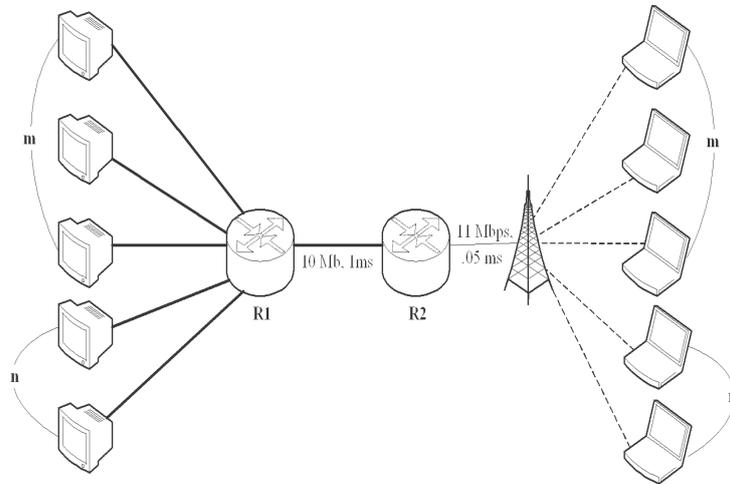


Figure 5: A Network Topology for Verifying Friendliness

Table 4: Friendliness Comparison of the EWR with WestWood+ (Throughputs in Kbps)

Number of Connections(New Reno)	Number of Connections(EWR)	Mean Throughput (New Reno)	Mean Throughput (EWR)
3	7	73.71	81.95
5	5	78.03	97.03
7	3	63.25	70.62

Finally, the friendliness of the proposed scheme with New Reno and Westwood+ has been tested by using a simple network topology shown in Figure 5. In simulations, there are 10 pairs of connections where  $m$  is the number of the hosts that communicate based on the proposed scheme and  $n$  is the number of hosts that use New Reno or Westwood+. Wireless link error rates have been set to vary between 0.1 and 10, and the corresponding results are summarized in Table 3 and Table 4. From the results, it has been shown that the proposed TCP EWR has good controllable friendliness compared with other two major TCP control schemes in error-prone wireless networking environment.

## 4 Conclusion

This paper introduces a new transmission control scheme for Future internetworking environment with error-prone wireless regions. The proposed scheme is designed to adjust its congestion window optimally based on the current network situation while estimating the available bandwidth at the receiver. As a result, accurate window update and transmission control can be possible to obtain high throughput while achieving satisfactory levels of fairness and friendliness, which are important indices for the proposed TCP EWR to be a feasible scheme.

## Acknowledgement

This research was supported by Kyung Hee University, Korea (20090724).

## Bibliography

- [1] Sumit Rangwala, Apoorva Jindal, Ki-Young Jang, and Konstantinos Psounis, *Understanding Congestion Control in Multi-hop Wireless Mesh Networks*, Proceedings of the 14th ACM international conference on mobile computing and networking, San Francisco, California, USA, pp. 291-302, 2008.
- [2] Dhiman Barman, Ibrahim Matta, Eitan Altman, and Rachid El Azouzi, *TCP Optimization through FEC, ARQ and Transmission Power Tradeoffs*, Lecture Notes in Computer Science, vol. 2957, pp. 87-98, 2004.
- [3] Rajashree Paul and Ljiljana Trajković, *Selective-TCP for Wired/ Wireless Networks*, in Proceedings SPECTS 2006, Calgary, AL, Canada, pp. 339-346, 2006.
- [4] Kai Xu, Ye Tian, and Nirwan Ansari, *TCP-Jersey for Wireless IP Communications*, IEEE Journal on Selected Areas in Communications, vol. 22, no. 4, pp.747-756, 2004.
- [5] Luigi A. Grieco and Saverio Mascolo, *Performance Evaluation and Comparison of Westwood+, New Reno and Vegas TCP Congestion Control*, ACM SIGCOMM Computer Communications Review, vol. 34, no. 2, pp. 25-38, 2004.
- [6] Jitendra Padhye, Victor Firoiu, Don Towsley, and Jim Kurose, *Modeling TCP Throughput: A Simple Model and its Empirical Validation*, ACM SIGCOMM Computer Communication Review, vol. 28, issue 4, pp. 303-314, 1998.
- [7] Cheng Peng Fu and Soung C. Liew, *TCP Veno: TCP Enhancement for Transmission Over Wireless Access Networks*, IEEE Journal on Selected Areas in Communications, vol. 21, no. 2, pp. 216-228, 2003.
- [8] Rajendra. K. Jain, Dah-Ming W. Chiu, and William R. Hawe, A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer System, DEC-TR-301, Eastern Research Lab., <http://www.cs.wustl.edu/~jain/papers/ftp/fairness.pdf>