Window-based Congestion Control for Wireless TCP

Hee-Jung Byun* and Jong-Tae Lim*

*Dept. of Electrical Engineering and Computer Science,
Korea Advanced Institute of Science and Technology
373-1, Guseong-dong, Yuseong-gu, Daejeon, 305-701, Korea.
(Tel: +82-42-869-3441; Fax: +82-42-869-3410; Email:jtlim@stcon.kaist.ac.kr)

Abstract: We propose a feedback-based congestion control algorithm for the wireless TCP network. In this paper, we present a new TCP protocol to control the congestion window size. In particular, the asymptotic analysis of the wireless TCP is presented. Through simulations, our algorithm shows an improvement of TCP's performance in wireless networks.

Keywords: wireless TCP, congestion control, RED, ECN

1. Introduction

TCP (Transmission Control Protocol) is the dominant protocol in the Internet, originally designed for the wired network where the link errors do not occur frequently. It is assumed that the congestion in the network is the primary cause for packet losses. However, in wireless networks, the packets are lost mainly because of the link error due to the channel fading, noise and interference. Therefore a TCP sender misunderstands the packet loss from this link error as the packet loss due to the congestion in the network. Hence, the TCP sender cannot differentiate between the packet loss due to the congestion and that due to the wireless link error, and thus reduces the congestion window size unnecessarily. Therefore TCP shows greatly degraded performance over the wireless networks.

Recently, several algorithms have been proposed to improve TCP's performance in the wireless networks [1]-[4]. The protocols proposed in [2]-[4] eliminate the packet loss due to the buffer overflows, so they only see the wireless losses. Therefore a TCP source only retransmits the lost packet on detecting the packet loss and adapts its window size using the feedback information.

In this paper, we propose an algorithm to improve TCP's performance in the wireless network. We use the IPv6 optional fields: round-trip propagation delay (RTPD) field and available bandwidth (ABW) field [3]-[4]. Using these information, we develop a modified window control algorithm to eliminate the congestion in the wired network and adapt the window size according to the wireless link state and the feedback information. Moreover, our proposed algorithm guarantees the fair sharing of the available bandwidth among the active connections and stabilizes the congestion window size and queue length at the desired value.

2. Congestion Window Control

2.1. Network Modeling

Consider a ECN capable TCP network with a single bottleneck link, where each connection has an identical propagation delay. By using the RTPD field and the ABW field in the IPv6's extended header [3]-[4], the round-trip propagation delay τ and the available bandwidth C/M are obtained, where C and M are the total bandwidth of the outgoing bottleneck link and the number of the active connections respectively. In addition, using the successive binary congestion information provided by ECN, we obtain the information of the queue length at the bottleneck link [6].

We propose a new TCP protocol with a receiver-side modification of the congestion window control scheme. With both RTPD and ABW conveyed to the TCP receiver, and the extracted queue length [6], the TCP receiver executes the congestion and flow control simultaneously as follows:

$$rwnd = \min(cwnd, rwnd)$$

where $cwnd$ is the adapted congestion window size using the information of RTPD, ABW and the queue length, and $rwnd$ is the receiver's advertised window size.

2.2. Proposed Algorithm

We consider the TCP packet flow from the fixed hosts to the mobile hosts as shown in Fig. 1. In order to define the network under consideration, we assume the followings:

- A receiver host updates its window size once per its round-trip time (RTT).
- The source traffic is modeled as a deterministic fluid flow.
- All connections have the same propagation delay.
- The outgoing link of the base station is the bottleneck link of the network.

Let M be the number of active connections routed through the bottleneck link. Provided that the waiting time of a packet at the base station is negligible, the system can be represented by a discrete-time model where a slot duration is equal to τ, which is the round-trip propagation delay. Let $w_i(n), 1 \leq i \leq M$ denote the window size of the sending host i at time n, and $q(n)$ denote the queue length at the base station at time n. Since each receiver host updates its window size every slot, the dynamics of the queue length [5] at the base station is described by

$$q(n+1) = q(n) + \sum_{i=1}^{M} w_i(n) - C[1-e(n)]\tau \qquad (1)$$

where $e(n)$ represents the process of packet loss ratio at time n on the wireless link. To stabilize the queue length at max_{th}

This work was supported by Institute of Information Technology Assessment under the fundamental research fund (sponsored by the Ministry of Information and Communication).

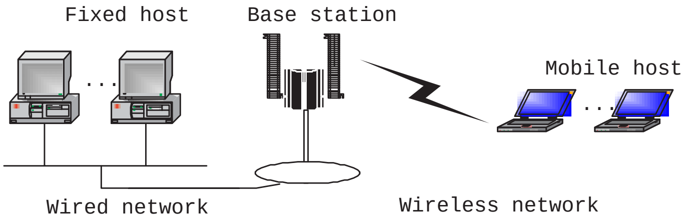

Fig. 1. Network model

and utilize the link fully, we introduce the following congestion window control scheme at the receiver host:

$$\begin{aligned} w_i(n) &= w_i^0 - \alpha(q(n-1) - max_{th}) \quad (2)\\ w_i^0 &= \frac{C}{M}\tau(1-p_e) \end{aligned}$$

where $p_e = E[e(n)]$, max_{th} is the the maximum threshold of the queue length, and α is the control gain to be chosen. Substituting (2) into (1), we obtain the following closed-loop equation:

$$\begin{aligned} q(n+1) &= q(n) + \alpha M(max_{th} - q(n-1)) \quad (3)\\ &\quad -C[p_e - e(n)]\tau \end{aligned}$$

Let $x(n) = \frac{q(n)}{q_c}$ where q_c is the total queue capacity. Then,

$$\begin{aligned} x(n+1) &= x(n) + \frac{\alpha M}{q_c}(max_{th} - x(n-1)\cdot q_c) \quad (4)\\ &\quad -\frac{C\tau}{q_c}[p_e - e(n)] \end{aligned}$$

Let $k_1 = \frac{\epsilon_a}{\epsilon}$, $k_2 = \frac{\epsilon_b}{\epsilon}$ where $\epsilon_a = \frac{\alpha M}{q_c}$, $\epsilon_b = \frac{C\tau}{q_c}$, and $\epsilon = \min(\epsilon_a, \epsilon_b)$. Since $q_c \gg 1$, without loosing much generality, we assume that $\epsilon \ll 1$. Then, (4) can be rewritten as

$$x(n+1) = x(n) + \epsilon\Phi(x(n-1), e(n)) \quad (5)$$

where

$$\Phi(x(n-1), e(n)) \triangleq k_1[max_{th} - x(n-1)\cdot q_c] - k_2[p_e - e(n)]$$

Since Φ can take arbitrarily large value but with negligibly small probabilities, (5) represents a slow-in-the average Markov walk process [8]-[9]. Let $z(n)$ denote the averaged value of $x(n)$. Applying the asymptotic theory for such processes [8]-[9], we obtain the following asymptotic approximation:

$$z(n+1) = z(n) + \epsilon(k_1[max_{th} - z(n-1)\cdot q_c]) \quad (6)$$

It is intuitively clear that if $\epsilon \to 0$, $|x(n) - z(n)| \to 0$. The equilibrium point of (6) is obtained from

$$z_s = \frac{max_{th}}{q_c}$$

Let $z(n)$ be the state vector by

$$\mathbf{z}(n) = [z(n) - z_s \;\; z(n-1) - z_s]^T$$

Then, (6) can be rewritten as follows

$$\mathbf{z}(n+1) = \mathbf{A}\mathbf{z}(n) \quad (7)$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & -\alpha M \\ 1 & 0 \end{bmatrix}$$

From the Jury's criterion, the system (7) is globally asymptotically stable if the control gain α is chosen as follows:

$$0 < \alpha < \frac{1}{M} \quad (8)$$

Consequently, the queue length converges to max_{th} and the window size of each connection converges to $\frac{C}{M}\tau(1-p_e)$. Therefore the congestion window control scheme guarantees that the network resources are fully utilized and each connection gets an equal share of the bandwidth. TCP connection adapts its window size using the feedback information to avoid the congestion-induced packet loss. Therefore when a packet loss occurs, the proposed algorithm treats it as an indication of the wireless link error. Accordingly, a TCP sender does not reduce the window size as a response to the packet loss.

3. Simulation Results

In this section, we describe the simulation results of our proposed algorithm. We compare our proposed algorithm with the modified ECN algorithm [2], which halves its window size when it receives an ECN mark, but only retransmits when it detects a packet loss through the duplicate acknowledgment. The simulation model is shown in Fig. 1. We use the following network parameters: The link speed of the base station C is 1500 [packet/s], max_{th} is 90 packets and q_c is 100 packets. The number of TCP connection M is 3, RTT of each TCP connection is 100ms, and the control gain α is chosen as $\alpha = 0.1$. Each source has an infinite data backlog to send and the packet losses by the wireless link error are occurred randomly with $p_e = 0.05$.

Figure 2 shows the congestion window size, the queue length and the average queue length at the base station for the modified ECN algorithm. In Fig. 2, the ECN algorithm shows that the window size and the queue length at the base station oscillate excessively because it halves its window size when it receives a mark or detects a loss through the timeout. Figure 3 illustrates the trajectories of the original system based on (5) and asymptotic approximation based on (6) in the proposed algorithm. As it follows from this figure, the approximated behavior follows the original behavior with acceptable tolerance. In addition, as shown in Fig. 3, the congestion window size obtained from the proposed algorithm shows the absolutely different behavior compared to Fig. 2, i.e., convergence is much faster without large oscillation. Moreover, the congestion window size converges to the fair window size and it remains constant at the steady state. Since each source transmits data packets based on the available bandwidth and the queue length at the base station, each connection gets an equal share of the bandwidth and the queue length keeps a level near the desired value (max_{th}). Hence, the proposed algorithm avoids the degradation of the throughput and utilizes the link capacity fully. We investi-

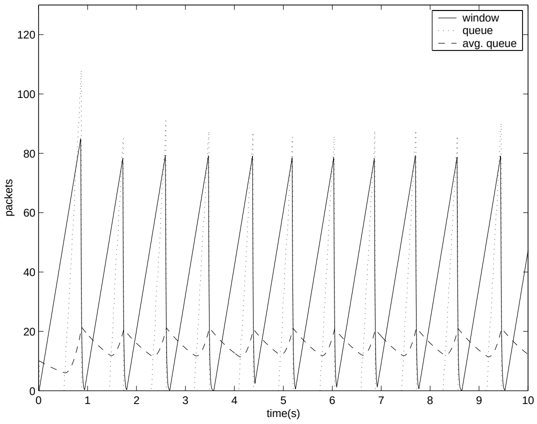

Fig. 2. Performance of ECN

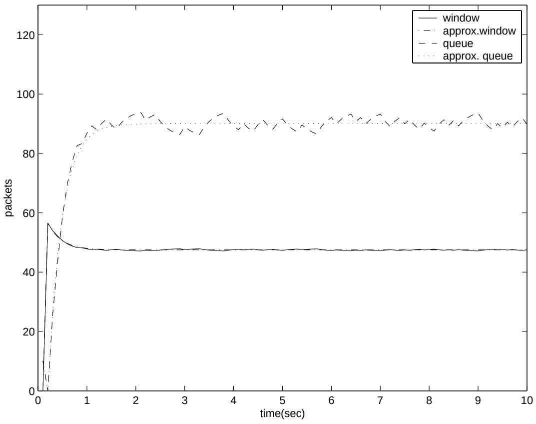

Fig. 3. Performance of proposed algorithm: trajectories of original system and asymptotic approximation

gate the impact of the number of the active connections for the proposed algorithm. In Fig. 4, as the number of connections increases, the mean value of each window size, i.e., $\sum_{n=1}^{100} w_i(n)/100$, $\forall i \in \{1, 10\}$ and the queue length are shown. As shown in Fig. 4, each congestion window size decreases as the number of the active connections increases, but the queue length keeps a level, max_{th} regardless of the number of active connections. This is because the proposed algorithm adapts the congestion window size based on the deviation of the queue length from the target max_{th} and the available bandwidth of the bottleneck link. Considering all the results, we conclude that the proposed algorithm significantly improves the overall wireless TCP performance.

4. Conclusion

We propose a new congestion window control algorithm for wireless TCP. The proposed algorithm avoids the congestion in the wired network and see the packet loss only due to the wireless link error. Our simulation results show that the proposed algorithm guarantees the fair sharing of the bandwidth among the active connections, and achieves high throughput and full utilization of the bottleneck link.

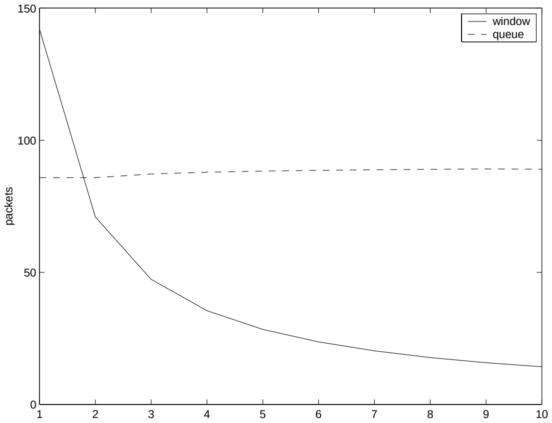

Fig. 4. Performance of proposed algorithm under various number of connections

References

[1] H. Balakrishnan, V. N. Padmanabhan, S. Seshan and R. H. Katz, “A comparison of mechanisms for improving TCP performance over wireless links,” *IEEE/ACM Trans. Networking*, vol. 5, pp. 756-769, 1997.

[2] S. Athuraliya, S. Low, V. H. Li and Q. Yin, “REM: active queue management,” *IEEE Networks*, vol. 5, pp. 48-53, 2001.

[3] M. Gerla, W. Weng and R. L. Cigno, “Bandwidth feedback control of TCP and real time sources in the Internet,” *IEEE GLOBECOM*, pp. 561-565, 2000.

[4] M. Gerla, W. Weng and R. L. Cigno, “BA-TCP: A bandwidth aware TCP for satellite networks,” *Computer Communications and Networks*, pp. 204-207, 1999.

[5] Y. Y. Kim and S. Q. Li, “Modeling fast fading channel dynamics for packet data performance analysis,” *IEEE INFOCOM*, pp. 1292-1300, 1999.

[6] H. J. Byun and J. T. Lim, “On window-based congestion control with explicit congestion notification,” *IEICE Trans. on Communications*, vol. E86-b, pp. 421-427, 2003.

[7] H. Ohsaki and M. Murata, “Steady state analysis of the RED Gateway: stability, transient behavior, and parameter setting,” *IEICE Trans. on Communications*, vol. E85-B, pp. 107-115, 2002.

[8] J. T. Lim, S. M. Meerkov and T. Zeng, “Simplified description of slow-in-the-average Markov walks,” *J. Math. analys and applic.*, vo. 158, pp. 476-486, 1991.

[9] J. T. Lim and K. H. Shim, “Asymptotic performance evaluation of Token Passing Networks,” *IEEE Trans. on Industrial Electronics*, vol. 40, pp. 384-385, 1993.

[10] V. Misra, W. B. Gong and D. Towsley, “Fluid-based analysis of a network of AQM routers supporting TCP flows with an application to RED,” *ACM SIGCOMM’00*, Sweden, pp. 151-160, 2000.

[11] Y. Hong, Y. Cao, H. Sun and K. S. Trivedi, “RED parameters and performance of TCP connections,” *Electronics letters*, vol. 37, pp. 1489-1491, 2001.