Journal of Institute of Control, Robotics and Systems (2010) 16(10):1013-1017
DOI:10.5302/J.ICROS.2010.16.10.1013 ISSN:1976-5622

관리 제어를 이용한 무선 TCP 성능 향상에 관한 방법

On Improving Wireless TCP Performance Using Supervisory Control

변 희 정*
(Heejung Byun[1])
[1]Suwon University

Abstract: This paper proposes a systematic approach to the rate-based feedback control based on the supervisory control framework for discrete event systems. We design the supervisor to achieve the desired behavior for TCP wireless networks. From the analysis and simulation results, it is shown that the controlled networks guarantee the fair sharing of the available bandwidth and avoid the packet loss caused by the buffer overflow of TCP wireless networks.

Keywords: wireless TCP, supervisory control

I. INTRODUCTION

When a wireless link forms a part of a network, the noncongestion losses may occur due to the wireless link error and TCP will half down its congestion window size unnecessarily. A number of algorithms have been proposed to improve TCP's performance over the wireless networks [1-10]. In [1-4], they decide whether packet losses are likely to be due to congestion or wireless link errors using the network congestion information or packet delay. The protocols proposed in [5,6] eliminate the packet loss due to the buffer overflows, so they consider the packet loss as due to the wireless link error on detecting the packet loss and only retransmit the lost packet. In [7], the bandwidth is controlled based on the radio conditions to enhance the wireless link capacity. The explicit window adaptation algorithms [8-10] suggest a congestion control scheme to explicitly inform the optimal window size to the TCP source by modifying the receiver's advertised window (AWND) field carried by TCP ACKs. In [11-13], they develop a variant of RCP (rate control protocol) under max-min criterion and study the local stability of RCP for a single resource with a large or small buffer. However, there is far less result on methods to provide fair sharing of the bandwidth. In addition, they are heuristic in nature and they do not present the formal methods to analyze their methods.

To deal with these problems, we present a formal method to develop a rate-based feedback control scheme over the wireless TCP networks based on the supervisory control framework [14,15]. This framework can effectively capture the dynamic behavior of discrete event systems and provide the existence conditions of a supervisor so that the controlled system meets a desired specification. However, there are still few control applications based on supervisory control due to its computational complexity on state space explosion. In this paper, we propose a discrete event model of TCP networks and design a simple supervisor to improve the wireless TCP's performance in terms of throughput and fairness with low computational complexity.

II. PROPOSED ALGORITHM

1. Supervisory control

We briefly describe the supervisory control framework [14,15]. In the supervisory control of discrete event systems, the system to be controlled is modeled by an automaton $G=(\Sigma, Q, \delta, q_0, Q_m)$ where Σ is the set of events, Q is the set of states, $q_0 \in Q$ is the initial state, $Q_m \subseteq Q$ is the set of marker states, and $\delta : \Sigma \times Q \mapsto Q$, the transition function, is a partial function defined at each state in Q for a subset of Σ. Let Σ^* denote the set of all finite strings (sequences) over Σ, including the empty string ε. A subset of Σ^* is called a language over Σ. The behavior of G is characterized by a language $L(G) := \{s \in \Sigma^* \mid \delta(s, q_0) \, is \;\; defined\}$, which is the set of event sequences generated in G.

To impose supervision on the system, we identify some of its events as controllable and the others as uncontrollable. The event set Σ is partitioned into controllable and uncontrollable events, i.e., $\Sigma = \Sigma_c \cup \Sigma_{uc}$. The controllable events in Σ_c can be disabled by a supervisor, while the uncontrollable events in Σ_{uc} are permanently enabled. A supervisor is then an agent which observes a sequence of events as it generated by G and enables or disables any of the controllable events.

To find out the existence of the supervisor, the following notion is necessary: Let us define K as the specification. Then a language $K \subset \Sigma^*$ is called *controllable* with respect to $L(G)$ if the following condition is satisfied [15]:

$$\bar{K}\Sigma_{uc} \cap L(G) \subseteq \bar{K}$$

where $\bar{K}$ is a prefix of K, i.e.,

$$\bar{K} := \{s \in \Sigma^* \mid (\exists t \in \Sigma^*) st \in K\}$$

This condition requires that extension of any prefix of K by an uncontrollable event that is feasible in the system should also be a prefix of K.

---

* 책임저자(Corresponding Author)

논문접수: 2010. 4. 6., 수정: 2010. 6. 28., 채택확정: 2010. 7. 12.

변희정: 수원대학교 IT대학 정보통신공학과(heejungbyun@suwon.ac.kr)

※ This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (grant number 20100022654).

2. Network modeling and window rate control

Consider the wireless TCP network where the base station is the bottleneck point of the network and M TCP connections are connected to the base station. Let $X = \lceil C \rceil$ where C is the capacity of outgoing link of the base station and $\lceil x \rceil$ is the smallest integer larger than x. Assume that the feedback rate (FR) is determined by the supervisor at the base station. The feedback rate is the available bandwidth for a connection and same to all connections routed through the base station. We partition the feedback rate into $(X+1)$ distinct elements, i.e., $FR \in \{0, 1, \cdots, X\}$.

To reflect the queue, we propose the free buffer function and the value of f_b is updated every T as follows:

$$f_b = \lambda(q^0 - q) \tag{1}$$

where $0 \le |f_b| \le R^{dev}_{\max}$, $R^{dev}_{\max}$ is the maximum increase/decrease of the feedback rate, q is the queue length of the base station, q^0 is the queue thresholds, and $\lambda(\lambda > 0)$ is a weighting constant to be chosen. Specifically, the value of T affects both the transient response and the control overhead. The choice of the update period T can be approached as a multicriteria optimization problem and the optimal update period can be calculated [16].

The basic idea of the proposed algorithm is to control the window rate based on the free buffer function in order to guarantee the fair sharing of the bandwidth and avoid the congestion. From this operation, our proposed algorithm can consider that the packet loss occurs due to the wireless link error by keeping the queue length at the desired queue length. To do this, the window size is controlled by a simple rule as:

$$w \leftarrow w + f_b$$

The window size increases as the free buffer length increases. On the contrary, the window size decreases as the free buffer length decreases. Since the window size is adapted every round-trip time, the network systems can be represented as the discrete event systems. Based on the supervisory control framework, we analyze the controlled system by capturing the dynamic behavior of TCP wireless networks and provide the existence conditions of a supervisor so that the controlled system meets our desired goal.

3. Supervisor design

In this section, we design the supervisor in order to guarantee the fair sharing of the bandwidth and avoid the congestion. After then, based on the supervisory control framework, we provide the existence conditions of a supervisor. Some of the notations used in this paper are listed in Table 1.

Let us introduce the event set Σ, the uncontrollable event set Σ_{uc}, and the set of states Q as

$$\Sigma = \{E_u, E_o, E_f, E^0_{JMP}, E^1_{JMP}, \cdots, E^X_{JMP}, E_{SUS}\}$$
$$\Sigma_{uc} = \{E_u, E_o, E_f\}$$
$$Q = \{R^0, \cdots, R^X, R^0_u, \cdots, R^X_u, R^0_o, \cdots, R^X_o, R^0_f, \cdots, R^X_f\}$$

Hence, an active event set at state R^i_u, R^i_o, R^i_f $(0 \le i \le X)$ is:

표 1. 표기법.
Table 1. Notations.

FR	Feedback rate
f_b	Free buffer function
E_u, E_o, E_f	Events representing that the value of f_b is updated and $f_b > 0, f_b < 0, f_b = 0$, respectively
E^i_{JMP} $(0 \le i \le X)$	Events representing the increase or decrease of the feedback rate (up-transition or down-transition to state R^i)
E_{SUS}	Event representing the maintenance of the current feedback rate
R^i	State representing that the feedback rate corresponds to i th element of FR and the value of f_b is waiting for the next update
R^i_u, R^i_o, R^i_f	State representing that the feedback rate is i th element of FR, the value of f_b is updated, and $q < q^0, q > q^0, q = q^0$, respectively.
BW_a	Available bandwidth for TCP flows at the base station

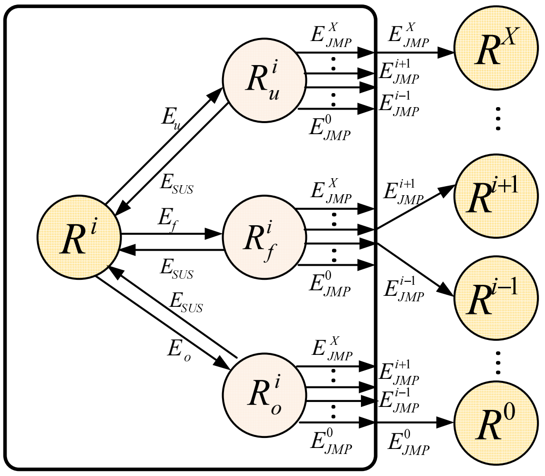

그림 1. 상태천이도.
Fig. 1. State transition diagram.

$$Act(R^i_u) = Act(R^i_o) = Act(R^i_f)$$
$$= \{E_{SUS}, E^0_{JMP}, \cdots, E^{i-1}_{JMP}, E^{i+1}_{JMP}, \cdots, E^X_{JMP}\}$$

The state transition diagram is shown in Fig. 1. When the value of f_b is updated at state R^i, one of uncontrollable events E_u, E_o, E_f is triggered. At state R^i_u, R^i_o, R^i_f, the supervisor adapts feedback rate by enabling or disabling controllable events according to the free buffer function. The objectives of the proposed method are to guarantee the fair sharing and avoid the congestion-induced packet loss by keeping the queue length at the desired queue length. The marked states represent the desired end-points of the system. Therefore, as shown in Fig. 2, the marked states are R^p and R^p_f where $p = BW_a / M$.

Let us define K_d as the specification to meet our objectives. We define

$$S = \{E_u E^i_{JMP} E_o E^j_{JMP}\}, (1 \le i \le X)(0 \le j \le X-1)$$

Then the specification K_d is described as

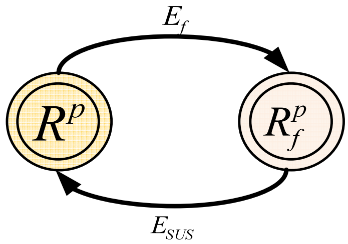

그림 2. 표기 상태.
Fig. 2. Marked state.

$$K_d = S^*(E_f E_{SUS})^+ \tag{2}$$

where $(X)^+ = XX^*$.

Let $K_1 := \{st \in \bar{K}_d \mid s \in \bar{K}_d, t \in \Sigma_{uc}\}, s_1 \in \bar{K}_d - K_1$ and $s_2 \in K_1$. Then obviously, $s_1 t \in \bar{K}_d$ and $s_2 t \notin \bar{K}_d$ for $t \in \Sigma_{uc}$. Moreover, if the queue length becomes the queue threshold, event E_u or event E_o cannot occur as long as the network parameters do not change. As it follows from the above, since $\bar{K}_d \Sigma_{uc} \cap L(G) \subseteq \bar{K}_d$, K_d is controllable with respect to $L(G)$.

The specification K_d means that the queue length should be the queue threshold after the transient behavior represented by S^*. Since K_d is controllable, we design a supervisor to achieve K_d by choosing the appropriate feedback rate.

Let $\Delta = \lceil |f_b| \rceil$. State R_u^i means that the network is under-utilized and so, the supervisor should increase the feedback rate by Δ over the current feedback rate (enable $E_{JMP}^{i+\Delta}$). On the contrary, since state R_o^i means that the network is over-utilized, the supervisor should decrease the feedback rate by Δ (enable $E_{JMP}^{i-\Delta}$). At R_f^i, the supervisor should enable E_{SUS} to hold the queue length at q^0. That is, the supervisor adapts the feedback rate according to the free buffer function so that the queue length converges to q^0. The control action of the supervisor is demonstrated in Table 2.

To convey the determined rate to the source, the proposed method uses the feedback signaling scheme presented in [8-10], i.e., the base station writes the feedback rate determined by the supervisor in the receiver's advertised window (AWND) field carried by the TCP acknowledgements (ACKs). Since TCP sources learn about the available bandwidth independent of packet loss and consider that the packet loss occurs due to the wireless link error by avoiding the buffer overflow, there is no need to reduce the congestion window following the packet loss. Furthermore, since the supervisor determines the feedback rate with the common buffer length, the proposed method does not require maintaining per-connection state at the base station. Therefore all connections routed through the base station will receive the same feedback rate.

표 2. 관리제어기 S.
Table 2. Supervisor S.

state	enabled event	condition
R_u^i	$E_{JMP}^{i+\Delta}$	$i \le (X - \Delta)$
	E_{JMP}^{X}	$(X - \Delta) < i < X$
	E_{SUS}	$i = X$
R_o^i	$E_{JMP}^{i-\Delta}$	$i \ge \Delta$
	E_{JMP}^{0}	$0 < i < \Delta$
	E_{SUS}	$i = 0$
R_f^i	E_{SUS}	-

When the specification K_d is achieved, i.e., $q = q^0$ in the steady state, the sum of incoming window rates (r_m) to the base station equals to BW_a:

$$\sum_{m=1}^{M} r_m = \sum_{m=1}^{M} FR^p = BW_a$$

where FR^p is the feedback rate when $q = q^0$. Since all connections routed through the base station receive the same feedback rate, we obtain the window rate of each connection:

$$r_m = FR^p = \frac{BW_a}{M}, m = 1, \cdots, M \tag{3}$$

That is, if K_d is achieved, the controlled network guarantees the fair sharing of the bandwidth, avoids the congestion, and considers that the packet loss occurs due to the wireless link error by keeping the queue length at the desired queue length. Therefore, the proposed supervisor can drive the network to reach the marked states.

III. SIMULATION RESULTS

We describe simulation results of our proposed window control algorithm. First, we compare the proposed method with TCP Veno [2] and XCP/RCP [12]. TCP Veno monitors the network congestion level and uses this information to decide whether packet losses are likely to be due to congestion or random bit error. The simulation model is shown in Fig. 3. We divide the connections into 3 Groups, which have $M_i (1 \le i \le 3)$ connections respectively. The round-trip propagation time of all the connections in Group i is $(100 \cdot i)$ ms. We use the following network parameters: The wired network is a LAN speed network and the base station is linked to the destinations via 10Mb/s noisy channel. The full buffer FTP traffic is generated and the packet size is 1Kbytes. We set $q^0 = 30$ Kbytes, $T = 100$ms, and $\lambda = 1$.

Fig. 4 shows the throughput with $M_1 = M_2 = M_3 = 1$. As shown in Fig. 4(a), TCP Veno has a large oscillation in the throughput and the fair sharing of the bandwidth is not achieved for a considerable period. Fig. 4(b) shows the throughput of

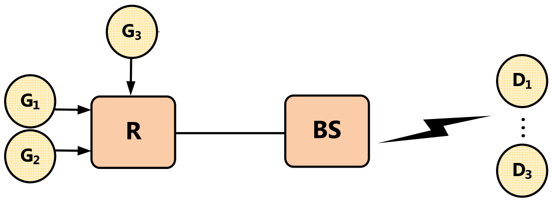

그림 3. 모의 실험 모델.
Fig. 3. Simulation model.

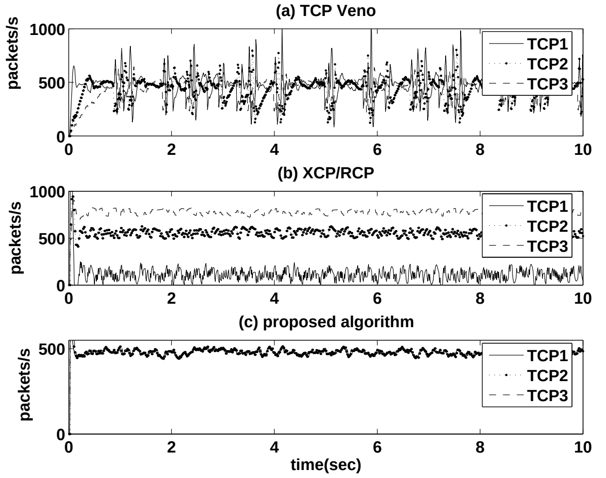

그림 4. 처리율: (a) TCP Veno (b) XCP/RCP (c) 제안 방법.

Fig. 4. Throughput: (a) TCP Veno (b) XCP/RCP (c) proposed method.

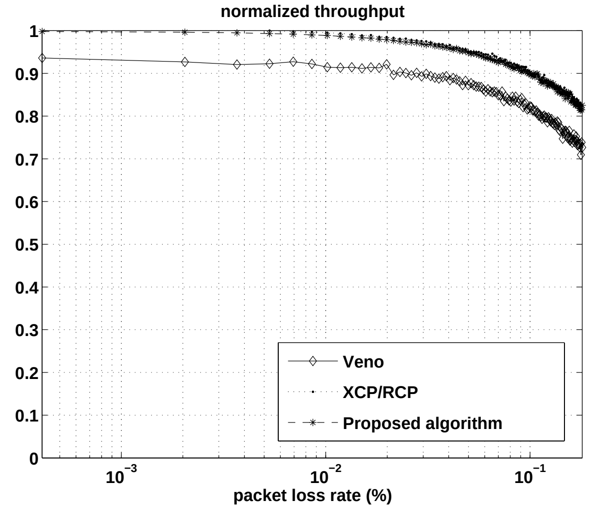

그림 5. 정규화 처리율과 패킷 손실율.

Fig. 5. Normalized throughput vs packet loss rate.

XCP/RCP. It shows the stable behavior but the fair sharing of bandwidth is not achieved among TCP connections. However, the proposed algorithm as shown in Fig. 4(c) guarantees the fair share of the bandwidth with rather small oscillation.

Fig. 5 shows the normalized throughput as the packet loss rate varies. For the proposed algorithm and XCP/RCP, the link is run close to capacity. Especially at the high error rate, the proposed and XCP/RCP method still have satisfactory throughput whereas Veno experiences degradation in the throughput. This is because the proposed supervisor and XCP/RCP method keep the queue length nonzero.

To show the fairness in the throughput, we measure the fairness index [4]. The values of fairness rage from 1/M to 1, with 1 corresponding to the best fair allocation among all connections. We consider multiple connections (1-10 for each Group). As shown in Fig. 6, XCP/RCP does not guarantee the fairness among TCP connections. TCP Veno and the proposed method achieve satisfactory fairness index, but the fairness index for the proposed method is higher than TCP Veno. This is because the proposed supervisor determines the feedback rate with the common buffer length and all connections routed through the base station transmit the packets with the same window rate. From the above results, we conclude that the controlled network significantly improves wireless TCP performance and guarantees the fair sharing of the available bandwidth.

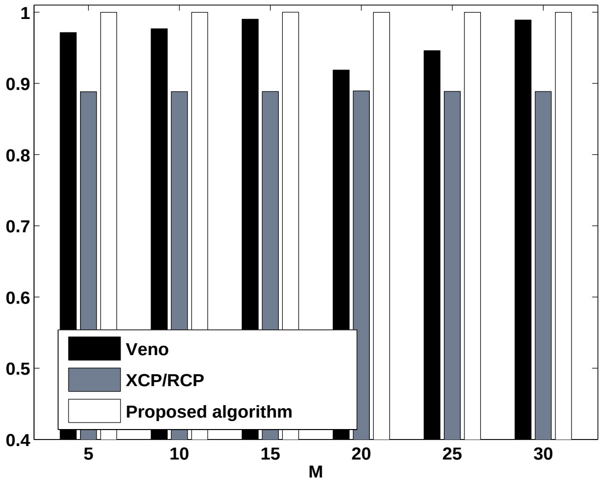

그림 6. 공정성 지수

Fig. 6. Fairness index.

IV. CONCLUSIONS

In this paper we propose the discrete event system approach to accommodate the rate-based feedback control scheme and make the supervisory control feasible for the network application. From the simulations, the controlled network has revealed better performance compared with the conventional schemes.

REFERENCES

[1] C. Liu and R. Jain, "Approaches of wireless TCP enhancement and a new proposal based on congestion coherence," *Proc. of the 36th Hawaii International Conference on System Sciences*, pp. 307-316, 2002.

[2] C. P. Fu and S. C. Liew, "TCP Veno: TCP enhancement for transmission over wireless access networks," *IEEE J. Select. Areas in Commun.,* vol. 21, no. 2, pp. 216-228, 2003.

[3] K. Xu, Y. Tian, and N. Ansari, "TCP-Jersey for wireless IP Communications," *IEEE J. Select. Areas in Commun*., vol. 22, no. 4, pp. 747-756, 2004.

[4] E. H. Wu and M. Z. Chen, "JTCP: Jitter-based TCP for heterogeneous wireless networks," *IEEE J. Select. Areas in Commun*., vol. 22, no. 4, pp. 757-766, 2004.

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

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

[7] J.-C. Moon and B.-G. Lee, "Rate-adaptive snoop: a TCP enhancement scheme over rate-controlled lossy links," *IEEE/ACM Trans. Networking*, vol.1, pp. 603-615, 2006.

[8] R. G. Mukhtar, S. V. Hanly, and L. L. H. Andrew, "Efficient internet traffic delivery over wireless networks," *IEEE Commun.*

Mag., vol. 41, no. 12, pp. 46-53, 2003.

[9] L. Kalampoukas, A. Varma, and K. K. Ramarkrishnan, "Explicit window adaptation: a method to enhance TCP performance," *IEEE/ACM Trans. Networking*, vol. 10, no. 3, pp. 338-350, 2002.

[10] T. Taleb, N. Kato, and Y. Nemoto, "An explicit and fair window adjustment method to enhance TCP efficiency and fairness over multihops satellite networks," *IEEE J. Select. Areas Commun.,* vol. 22, no. 2, pp. 371-387, 2004.

[11] F. Kelly, G. Raina, and T. Voice, "Stability and fairness of explicit congestion control with small buffers," *ACM SIGCOMM Computer Communication Review*, vol. 38, issue 3, pp. 51-62, 2008.

[12] H. Balakrishnan, N. Dukkipati, N. McKewon, and C. Tomlin, "Stability analysis of explicit congestion control protocols," *IEEE Communications Letters*, vol. 11, no. 10, pp. 823-825, 2007.

[13] T. Voice and G. Raina, "Stability analysis of a max-min fair rate control protocol (RCP) in a small buffer regime," *IEEE Trans. Automatic Control*, vol. 54, no. 8, pp. 1908-1913, 2009.

[14] P. J. Ramadge and W. M. Wonham, "Supervisory control of a class of discrete event processes," *SIAM J. Control and Optimization*, vol. 25, no. 1, pp. 206-230, 1987.

[15] P. J. Ramadge and W. M. Wonham, "The control of discrete event systems," *Proceedings of the IEEE*, vol. 77, no. 1, pp. 81-98, 1989.

[16] L. Benmohamed and S. M. Meerkov, "Feedback control of congestion in packet switching networks: the case of a single congested node," *IEEE/ACM Trans. Networking*, vol. 1, no. 6, pp. 693-708, 1993.

변 희 정

1999년 숭실대학교 정보통신공학과 학사. 2001년 한국과학기술원 전자전산학과 석사. 2005년 한국과학기술원 전자전산학과 박사. 2010년 삼성종합기술원, 삼성전자. 2010년~현재 수원대학교 IT대학. 관심분야는 네트워크 제어.