C-Snoop: Cross Layer Approach to Improving TCP Performance over Wired and Wireless Networks

Jaehoon Kim and Kwangsue Chung

School of Electronics Engineering, Kwangwoon University, Korea

Summary
TCP continues to be the most important transport layer communication protocol. In heterogeneous wired and wireless networks, however, the high packet loss rate over wireless links can trigger unnecessary execution of TCP congestion control algorithms, resulting in performance degradation. Several solutions have been proposed to address the known problems that TCP faces when running over wireless networks. Of those solutions, the localized link layer schemes, such as Snoop, SACK-Aware-Snoop and SNACK, has been shown to be the most effective. However in the wireless channel with high packet loss rate, these mechanisms do not work well. In this paper, we propose a new local retransmission scheme based on cross layer approach, called Cross-layer Snoop(C-Snoop) protocol, to solve the limitation of existing localized link layer schemes. From the simulation result, C-Snoop is proved to better TCP throughput and energy efficiency than existing mechanisms.
Key words:
Cross layer, TCP, Snoop, SNACK, Energy efficiency

1. Introduction

Since its inception 30 years ago, the Transmission Control Protocol (TCP) has grown to be the most important communication protocol and responsible for the stability of the Internet. However, when TCP works over wireless environments several well-known problems affect its performance because it is tuned to perform well in traditional networks where congestion is the primary cause of packet loss. In wireless networks, packets are lost due to high Bit Error Rates (BERs), signal fading, user mobility, hand-off procedures, channel asymmetries, and others, and not due to network congestion. As a result, TCP misinterprets these losses to be due to congestion and applies its congestion control algorithms unnecessarily, yielding low throughputs [1].

Several performance enhancing solutions have been proposed to help TCP differentiate congestion related losses from wireless losses. These solutions have been proposed at various layers of the protocol stack and can be mainly classified as link layer mechanism, transport layer mechanisms, and also newer versions of TCP [2~8].

Among the above mechanisms, the Snoop protocol has been shown to be the best performing solution when the packet loss rate is high[9]. However, Snoop can only provide single packet loss information within one local RTT (Round-Trip-Times). Under high loss rate wireless network environment, Snoop does not work well because it mimics the TCP error recovery mechanism, which is not very robust under harsh error conditions.

To solve this problem, SACK-Aware-Snoop and Selective Negative Acknowledgment (SNACK) mechanism have been proposed[10],[11]. However in the wireless channel with high packet loss rate, SACK-Aware-Snoop and SNACK mechanism do not work well because of two reasons: (a) end-to-end performance is degraded because duplicate ACKs themself can be lost in the presence of bursty error, (b) energy of mobile device and bandwidth utilization in the wireless link are wasted unnecessarily because of additional TCP header option such as SACK or SNACK in the wireless link.

In this paper, we propose a new local retransmission mechanism based on the cross layer approach, called Cross-layer Snoop(C-Snoop) protocol, to solve the limitation of the existing localized link layer mechanisms [12],[13]. C-Snoop protocol includes caching lost data packet and performing local retransmission based on a few policies dealing with MAC-layer's packet delivery information and local retransmission timer. From the simulation result, we could see more improved TCP throughput and energy efficiency than the existing localized link layer mechanisms.

The remaining of the paper is organized as follows. Section 2 reviews past works in this area. The following section outlines the principles of our C-Snoop protocol. The subsequent section presents the simulation results, and lastly we conclude the work.

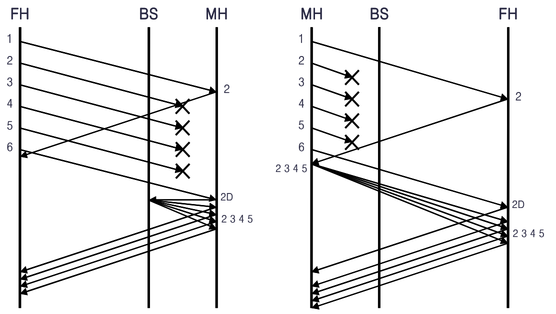
2. Related Work

When Snoop recovers multiple losses in wireless link, it consumes many local RTT, which occurs the TCP's

Manuscript received March 5, 2007
Manuscript revised March 25, 2007

retransmission timeout and then it goes into slow start. Therefore transmission performance is significantly degraded[10],[11]. To solve this problem, the SNACK mechanism has been proposed. Like TCP-SACK, the SNACK also uses the additional TCP header option. SNACK is designed to recover multiple packet losses within one local RTT through several loss blocks in the SNACK header option.

The SNACK proposes two protocol components, called SNACK-Snoop and SNACK-TCP, to effectively recover multiple losses in wireless link. From FH to MH, SNACK-Snoop performs the functions of detecting wireless multiple losses and piggybacking the SNACK information, while SNACK-TCP performs the functions of processing ACKs with SNACK information and retransmitting the losses within one local RTT. From MH to FH, SNACK-Snoop performs the functions performed by SNACK-TCP in the direction from FH to MH. On the other hand the SNACK-TCP performs the function performed by SNACK-Snoop in direction from MH to FH. Figure 1 shows the cooperation of the two protocol components to recover from four continuous packet losses in both directions.



(a) From FH to MH (b) From MH to FH
Fig. 1 Recovery from four drops in both directions.

In this manner, SNACK effectively recovers multiple packet losses in wireless link. However, SACK also has a few limitations in the wireless channel with high packet loss rate due to two reasons as follows:

i) Like Snoop, SNACK recovers packet losses through the ACK packet of transport layer(TCP-ACK). This technique degrades end-to-end transmission performance due to duplicate acknowledgment themselves can be lost in the presence of bursty error. Furthermore, this technique offers great improvement in the model of wired-cum-wireless networks. But when used in wireless-cum-wired, it is regarded as ineffective because, in reverse direction, TCP-ACK packets are returned too late for local error recovery [14].

ii) SNACK mechanism uses redundant SNACK packet under the harsh error condition. This technique is not only modifying existing standard mechanism at BS and MH respectively but wasting the bandwidth utilization and inefficiently consuming energy of mobile device. The energy efficiency of mobile device is important part of wireless networks due to its limited battery power.

3. C-Snoop Protocol

In this Section, we propose C-Snoop protocol that is a new cross layer approach to solve the limitation of the existing localized link layer mechanisms. The considered scenario is a WLAN employing the IEEE 802.11 protocol at the MAC and physical layers. The Distributed Coordination Function(DCF) is assumed to be employed to discipline access on the wireless channel.

3.1 System Architecture for C-Snoop

IEEE 802.11 provides reliable link layer data transmission by handling packet delivery problem through MAC layer's acknowledgement(MAC-ACK). Therefore, the MAC layer detects packet losses at the first time in wireless networks. The purpose of C-Snoop running on BS and MH is to perform efficiently local retransmission through collaboration between MAC layer, equipped with 802.11 protocol, and IP layer. For efficient local retransmission, C-Snoop retransmits lost packet in wireless link in case of the MAC-ACK received from the receiver(MH or BS) that is equal to the receiver of lost packet or a novel local retransmission timer expired at the sender(MH or BS).

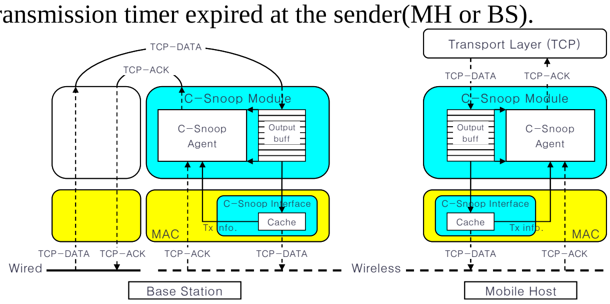


Fig. 2 System architecture for C-Snoop

Figure 2 shows the general architecture of BS and MH with a cross layer approach. To solve the limitation of the existing localized link layer mechanisms, additional two modules, called C-Snoop Module(IP layer) and C-Snoop Interface(MAC layer), are inserted at the IP and MAC layer. The advantage of the C-Snoop protocol's architecture is to avoid any change at the IP and MAC layers and operate the local recovery mechanism independently at both BS and MH by using own MAC-layer information. Because of C-Snoop's independent loss recovery mechanism, both From MH to FH and From FH to MH transmission performance can be improved. In addition, even though either BS or MH does not support the C-Snoop, the performance of C-Snoop supported direction can be still improved. For an example, even if BS does not support C-Snoop protocol, From MH to BS transmission performance can be enhanced.

We will discuss about the detailed operation of C-Snoop agent's interaction with other layers and novel retransmission timer in section 3.2 and 3.3.

3.2 C-Snoop Agent Interaction with MAC Layer

The MAC layer detects packet losses at the first time in wireless networks. To recover quickly from bursty loss, the C-Snoop agent interacts with the C-Snoop Interface. The C-Snoop Interface is inserted at MAC layer in order to detect the packet losses as early as possible in wireless networks and provide information about the packet delivery to the destination host to C-Snoop agent. For this purpose, two events are specified in C-Snoop Interface:

i) DELIVERED event: for the indication of a successful packet delivery. This event is generated upon the reception of MAC-ACK at the MAC layer as the indication that a data packet is successfully received by the destination node.

ii) UNDELIVERED event: for the notification that the MAC layer is not able to deliver the packet. This event is generated when the timeout at the MAC layer is triggered.

3.2 C-Snoop Agent Interaction with Wired Networks or Transport Layer

Whenever a TCP data packet is received, C-Snoop stores the relevant information, including queuing delay for local retransmission timer, and caches the packet to local buffer, while the packet itself gets through to the lower layers. And then, C-Snoop agent remains waiting for an event from the C-Snoop Interface which will inform either the successful or unsuccessful packet delivery.

In case of DELIVERED event, C-Snoop agent removes the packet stored in local buffer and retransmits previously lost packets which have the same destination address with higher priority. The local retransmission timer is also updated by transmission delay and queuing delay. In case of UNDELIVERED event, C-Snoop agent will handle the lost packet to local retransmission bounded in novel local retransmission timer.

Figure 3 shows the local recovery procedure of C-Snoop after the packets 2 to 4 are dropped. Like Figure 3, C-Snoop agent is caching the TCP data packet received from wired networks or transport layer and then recovering previously lost packet through C-Snoop Interface's DELIVERED event.

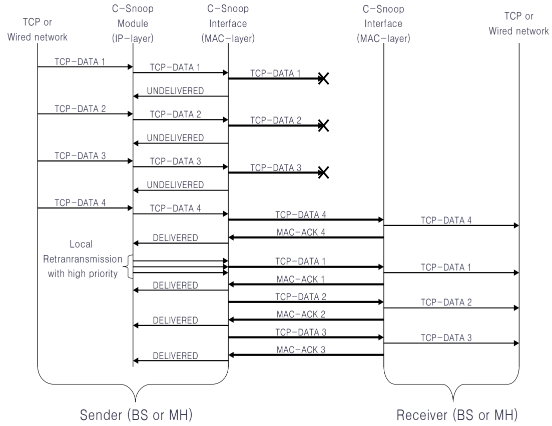


Fig. 3 Recovery procedure based on MAC-ACK

3.3 Local Retransmission Timer for C-Snoop Agent

In addition to retransmitting packets depending on the number and type of acknowledgments received, the existing localized link layer mechanisms also perform retransmissions driven by timeouts. Those mechanisms trigger the timeout only after the first retransmission of a packet from the cache, caused by the arrival of a duplicate acknowledgment. The ensures that a negligible number of unnecessary retransmissions occur for packets that have already reached at MH[2],[15]. However, the mechanism degrades the transmission performance in case that sender's transmission window size is too small in wireless channel with bursty loss. Figure 4 shows the inability of the existing local retransmission timer. If all data and ACK packet are lost in wireless channel, TCP at FH faces a retransmission timeout and goes into slow start, which significantly reduces the throughput.

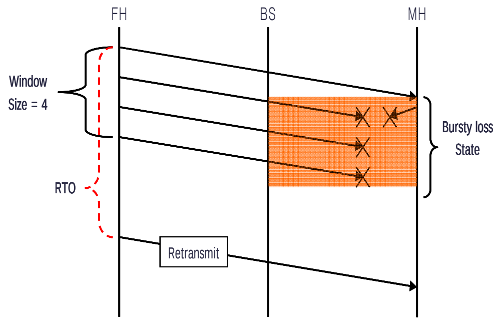


Fig. 4 Problem in existing retransmission timer

C-Snoop is adding local retransmission timer to the lost packet, when C-Snoop agent received explicit packet loss information, that is, UNDELIVERED event from C-Snoop Interface. Thus, C-Snoop ensures that unnecessary retransmissions are not occurred. Figure 5 shows the queuing delay(Q_d) and transmission delay(T_d) for computing the round-trip time of the wireless link(W_{RTT}).

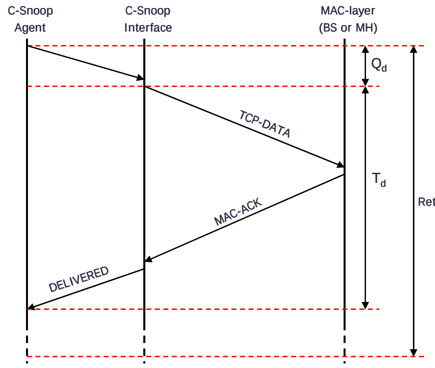


Fig. 5 RTT of the wireless link

In order to retransmit the lost packet, C-Snoop agent defines, the local retransmission timer derived in Eq.(1). C-Snoop agent adds this timer to cached packet for the local retransmission. When this timer is expired, the lost packet is retransmitted without MAC-ACK received.

$$
\begin{aligned}
&W_{RTT} = Q_d + T_d \\
&SW_{RTT} = (1-\alpha)\times SW_{RTT} + \alpha \times W_{RTT} \\
&RW_{RTT} = (1-\beta)\times RW_{RTT} + \beta \times |SW_{RTT} - W_{RTT}| \\
&Local\ retransmission\ timer = SW_{RTT} + 4RW_{RTT}
\end{aligned}
\quad (1)
$$

C-Snoop's timer prevents unnecessary retransmissions because it is just applied to lost packets through the UNDELIVERED event of C-Snoop Interface. In addition, C-Snoop retransmits lost packet in wireless link before TCP's retransmission timer expired due to the calculation manner of C-Snoop's timer is very similar to that of transport layer's retransmission timer(RTO) and C-Snoop's timer is calculated based on round-trip delay in wireless link(W_{RTT}). Figure 6 shows the local recovery step of C-Snoop based on local retransmission timer.

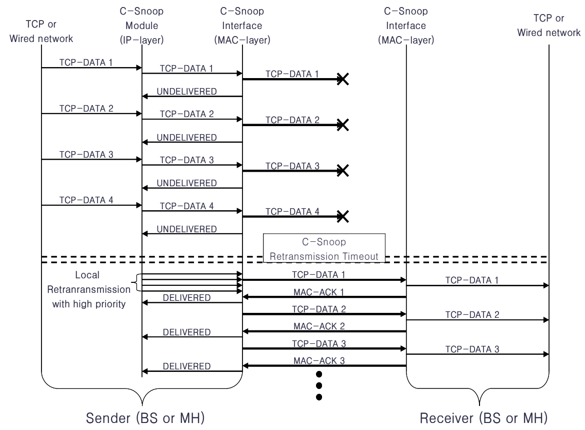


Fig. 6 Recovery procedure based on the retransmission timer in C-Snoop

4. Performance Evaluation

In this Section we evaluate the C-Snoop protocol based on the ns-2 simulator[16]. This evaluation has been carried out to show some improvements on the throughput and the energy efficiency of C-Snoop in experimental networks. The performance of C-Snoop protocol is compared with that of Snoop and SNACK mechanisms.

4.1 Simulation Environment

The network configuration for the simulation is shown in Figure 7. Since we focus on performance in presence of bursty error state, the node mobility is not considered in our simulation. The MH is based on IEEE 802.11b with 11Mbps. Simulation parameters are set to satisfy the IEEE 802.11b specification at both physical and link layers.

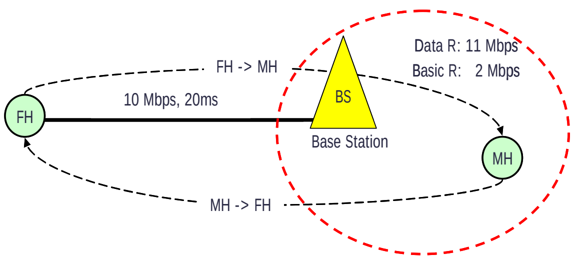


Fig. 7 Network configuration for simulations

It is well known that losses in wireless channels usually occur in a bursty fashion. These losses can be modeled as a two-state link error model consisting of a good state and a bad state. To generate the bursty error, the wireless link drops compulsively the packets as the packet loss rate. The ranges of burst loss rate is 0~10%. The source node(FH or MH) sends continuously packet to the destination node(FH or MH). The initial energy of the MH set to 100J(Joule). All nodes consume the 0.6W(Watt) for transmitting a packet and 0.3W for receiving a packet. In case of the SNACK mechanism, when packet loss has occurred in wireless link, energy consumption rate becomes grow in proportion of the size of ACK packet because the receiver (MH or BS) sends ACK packet with redundant data bit.

4.2 Simulation Results

The local recovery of C-Snoop is unaffected by the TCP-ACK packet lost. C-Snoop can also recover multiple packet losses faster than existing localized link layer mechanism. Therefore, C-Snoop performs better than SNACK mechanism. Figure 8, 9, 10 and 11 show the throughput and sequence number of SNACK and C-Snoop protocol at 5% burst packet loss rate.

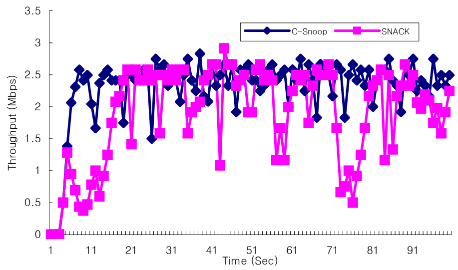


Fig. 8 Throughput at 5% Burst Packet Loss Rate (From FH to MH)

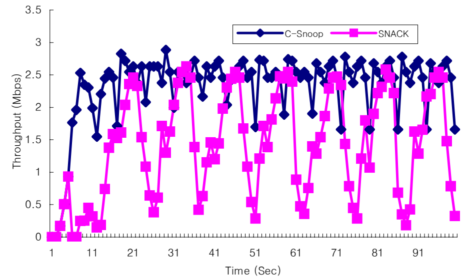


Fig. 9 Throughput at 5% Burst Packet Loss Rate (From MH to FH)

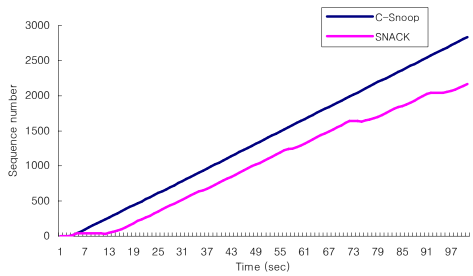


Fig. 10 Sequence Number at 5% Burst Packet Loss Rate (From FH to MH)

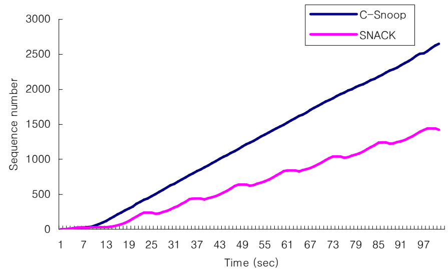


Fig. 11 Sequence Number at 5% Burst Packet Loss Rate (From MH to FH)

At 0%~10% burst packet loss rate, the performance of C-Snoop protocol is better than existing localized retransmission mechanisms in regardless of the transmission direction as shown in Fig.12 and 13. Moreover since each C-Snoop Interface of the MH and BS can provide explicit wireless loss information to C-Snoop agent by itself, both the FH to MH and MH to FH transmission performance is similarly improved.

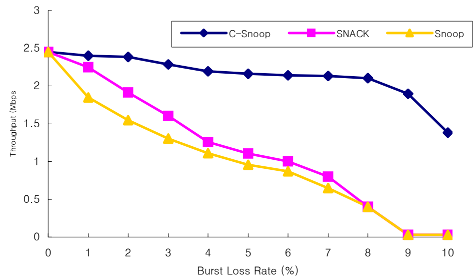


Fig. 12 Average Throughput Vs. Burst Packet Loss Rate (From FH to MH)

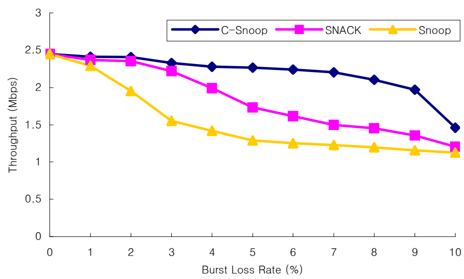


Fig. 13 Average Throughput Vs. Burst Packet Loss Rate (From MH to FH)

We also investigated the energy efficiency of each protocol in a bursty loss state. The energy efficiency of C-

Snoop, SNACK and Snoop are shown in Figure 14. The energy efficiency can be evaluated on the basis of Eq.(2).

$$Energy\ \ Efficiency(\eta) = \frac{Throughput}{Consumed\ \ Energy}(Kb/sJ) \quad (2)$$

During the whole simulation time, 100 seconds, C-Snoop protocol accomplishes better energy efficiency than Snoop and SNACK. The improvement is approximately about 20%~90%. At 5% packet loss rate, the throughput of C-Snoop, SNACK and Snoop are respectively about 0.9 Mbps, 1.2 Mbps and 2.1 Mbps. And the consumed energies are respectively about 26 Joules, 31 Joules and 39 Joules. Based on the simulation result, it is proved that Snoop has the poor energy efficiency with the low throughput and high consumed energy. It means that Snoop retransmit the large amount of packets. The consumed energy of C-Snoop is similar to that of SNACK but the throughput of C-Snoop is much higher than that of SNACK. Therefore, C-Snoop has the better energy efficiency than SNACK. This result shows that C-Snoop's retransmission mechanism recover burst loss in wireless link faster than SNACK due to its MAC-layer's explicit loss information and efficiency local retransmission timer.

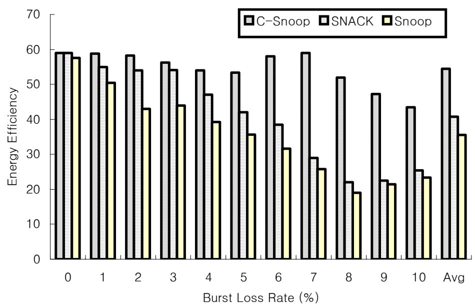


Fig. 14 Energy Efficiency Vs. Burst Packet Loss Rate

5. Conclusion

This paper has proposed a novel protocol called C-Snoop to solve the problem of existing localized link layer mechanisms. The key idea of the C-Snoop is to introduce the capability to detect bursty losses at the BS and MH in a wireless link, and to provide MAC-layer's explicit loss information to the C-Snoop agent in a speedy manner to trigger immediate retransmissions for packet lost in the wireless link. The C-Snoop protocol can provide explicit loss information and efficiency local retransmission timer at BS and MH. Through changing the functions deployed by the two protocol components, namely C-Snoop Module and C-Snoop Interface, both the MH to FH and FH to MH transmission performance can be greatly enhanced. Our analyses and simulation results show that C-Snoop can effectively enhance TCP performance over wireless links, particularly in those wireless networks with high packet loss rates and serious bursty losses. In the future, we will focus our attention on the extension of C-Snoop protocol such as handoff support. In addition, we will consider different types of wireless links such as cellular networks.

Acknowledgment

This work was supported by the Korea Science and Engineering Foundation(KOSEF) grant funded by the Korea government(MOST) (**R01-2005-000-10934-0**)

References

[1] V. Tsaoussidis and I. Matta, "Open Issues on TCP for Mobile Computing," The Journal of Wireless Communications and Mobile Computing, pp. 3-20, February 2002.
[2] H. Balakrishnan, S. Seshan, and R.H Katz. "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks," ACM Wireless Networks, pp. 469-481, December 1995.
[3] N. Vaidya, M. Mehta, C. Perkins and G. Montenegro, "Delayed Duplicate Acknowledgements: A TCP-Unaware Approach to Improve Performance of TCP over Wireless," Technical Report, Computer Science Dept., Texas A&M University, 1999.
[4] C. Parsa and J. Garcia-Luna-Aceves, "Improving TCP Performance over Wireless Networks at the Link Layer," Mobile Network and Applications, 2000.
[5] A. Bakre and B. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts," Proceedings of International Conference on Distributed Computing System, 1995.
[6] K. Brown and S. Singh, "M-TCP: TCP for Mobile Cellular Networks," ACM SIGCOMM Computer Communication Review, pp. 19-43, October 1997.
[7] H. Balakrishnan and R. Katz, "Explicit Loss Notification and Wireless Web Performance," Proceedings of GLOBECOM, November 1998.
[8] B. Bakshi, N. Vaidya and D. Pradhan, "Improving the Performance of TCP over Wireless Networks," Proceedings of 17th International Conference on DCS, pp. 365-373, July 1997..
[9] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. H. Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links," ACM Transactions on Networking, pp. 756-769, December 1997.
[10] S. Vagala and M. Vehicular, "The TCP SACK-aware Snoop Protocol for TCP over Wireless Networks," Proceedings of Vehicular Technology Conference, pp. 2624-2628, October 2003.
[11] F. Sun, Li. V.O.K. and S. Liew, "Design of SNACK Mechanism for Wireless TCP with New Snoop," Proceedings of IEEE Wireless Communications and Networking Conference, pp. 1051-1056, March 2004. .

[12] V. Raisinghani and S. Iyer, "Cross-layer Design Optimization in Wireless Protocol Stack," Computer Communications, pp. 720-724, May 2004.
[13] V. Raisinghani and S. Iyer, "ECLAIR: An Efficient Cross Layer Architecture for Wireless Protocol Stacks," Fifth World Wireless Congress, May 2004.
[14] G. Xylomenos and G. Polyzos, "Quality of Service Issues in Multi-service Wireless Internet Links," Proceedings of The International Workshop on QoS in Multi-service IP Networks, pp. 347-364, 2001
[15] K. Ratnam and I. Matta, "WTCP: An Efficient Mechanism for Improving TCP Performance over Wireless Links," Proceedings of Third IEEE symposium on Computer and Communications, pp. 74-78, June 1998.
[16] The network simulator ns-2, http://www.isi.edu/nanam/ns/

Jaehoon Kim received the B.S. degree from Kwangwoon University, Seoul, Korea, from Electronics and Communications Department in 2003. Currently he is pursuing M.S. degree in Electronics & Communications Department at Kwangwoon University under the guidance of Prof. Kwangsue Chung. His research interests include communication protocols and networks and embedded & distributed system.

Kwangsue Chung received the B.S. degree from Hanyang University, Seoul, Korea, M.S. degree from KAIST (Korea Advanced Institute of Science and Technology), Seoul, Korea, Ph.D. degree from University of Florida, Gainesville, Florida, USA, all from Electrical Engineering Department. Before joining the Kwangwoon University in 1993, he spent 10 years with ETRI (Electronics and Telecommunications Research Institute) as a research staff. He was also an adjunct professor of KAIST from 1991 to 1992 and a visiting scholar at the University of Califonia, Irvine from 2003 to 2004. His research interests include communication protocols and networks, QoS mechanism, and video streaming. Dr. Chung is a senior member of IEEE.