Journal of The Institute of Electronics and Information Engineers Vol.52, NO.8, August 2015 http://dx.doi.org/10.5573/ieie.2015.52.8.010
ISSN 2287-5026(Print) / ISSN 2288-159X(Online)

논문 2015-52-8-2

위성 통신 시스템에서 TCP 연결 분할 기반 PEP의 성능 평가

(Performance Evaluation of PEP based TCP Splitting Scheme in Satellite Communication Systems)

네 띠 넬*, 이 규 환*, 김 종 무*, 김 재 현**

(Nathnael Gebregziabher Weldegiorgis, Kyu-Hwan Lee, Jong-Mu Kim, and Jae-Hyun Kim©)

Abstract

A satellite communication system is one of viable solutions for Internet applications running in wide areas. However, the performance of TCP can be seriously degraded in the satellite networks due to long round-trip time (RTT) and high bit error rate (BER) over satellite links. Therefore, a performance enhancing proxy(PEP) based TCP splitting connection scheme is used in the satellite link to improve the TCP performance. In this paper, we implement PEP testbed and conduct experiment to evaluate the performance of TCP splitting connection by comparing with high-speed TCP solutions in various environments. In our experimental environment, we consider multiple connections, high packet loss, and limited bandwidth. The experiment results show that PEP improves the TCP throughput than high-speed TCP variants in various environments. However, there is no improvement of the TCP throughput with the limited bandwidth because there is packet loss caused by both the congestion and the channel error.

요 약

위성 통신 시스템은 광범위한 영역에서 TCP/IP를 통해 실행되는 인터넷 애플리케이션에 대한 합리적인 해결책중 하나이다. 그러나, 위성 통신에서의 긴 왕복전송 시간(RTT)과 높은 비트 에러율(BER)로 인해 TCP프로토콜은 심각한 성능 저하가 발생한다. 그러므로 위성 링크에 PEP를 적용하여 위성 환경에서의 TCP의 성능을 향상시킨다. 본 논문에서는 PEP 테스트베드를 구현하고, 다양한 환경에서 무선 채널을 고려한 변종 고속 TCP 기법과 TCP 연결 분할의 성능을 비교 평가하는 실험을 다중 연결, 높은 패킷 손실 그리고 제한된 대역폭에서 수행하였다. 성능 분석 결과 PEP는 다양한 환경에서 변종 고속 TCP보다 TCP 처리율을 더 많이 향상할 수 있음을 확인하였다. 하지만 네트워크 혼잡이 포함된 환경에서는 다른 변종 TCP와 비슷한 성능을 나타냈다.

Keywords : PEP, BER, TCP-splitting, High-speed TCP protocols, PER, E2E

* 학생회원, 아주대학교 무선인터넷 연구실
(Ajou University)
** 평생회원, 아주대학교 무선인터넷 연구실
(Ajou University)
© Corresponding Author(E-mail: jkim@ajou.ac.kr)
※ 이 논문은 2014년도 정부(미래창조과학부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(NRF-2014R1A2A2A01002321)
Received ; April 30, 2015 Revised ; July 7, 2015
Accepted ; August 1, 2015

Ⅰ. Introduction

Satellite communication systems represent a viable solution for Internet access for wide areas. New generation satellite system architectures are being designed to be fully IP based. Therefore, they are an ideal mean for offering TCP/IP based Internet services over long distances to achieve a reliable end-to-end (E2E) connection for data delivery. However, a link based geostationary earth orbit (GEO) satellite channel induces long propagation delay and high packet error rate (PER) due to random and dynamic wireless channel. The combination of long round-trip times (RTT) and high packet loss rates (PER) over these links create an environment that seriously degrades the performance of TCP on the satellite links[1~3].

High-speed TCP protocols like CUBIC[4], Hybla[5], and Compound TCP[6] have been proposed to address issues in wireless links. They use packet losses and increase of queuing delay as indicators of congestion. These typically involve tuning TCP so that the long RTTs satellite links do not negatively impact performance. However, they severely degrade their performance in the presence of high bit error rate(BER)[7].

A well-known solution to these problems is performance enhancing proxy (PEP). It is a network agent that improves the network performance between E2E nodes. There are many researches on PEP to enhance the TCP performance in the satellite communication[8~10]. PEP with TCP splitting connection is usually employed to partition an E2E connections in order to isolate dissimilar networks from them.

In this paper, we conduct experiment to evaluate the performance of PEP over satellite link using our implemented testbed. The rest of the paper is organized as follows: in section Ⅱ, we describe the related works on TCP performance over satellite communication. Section Ⅲ presents the system model. Testbed description of PEP is provided in section Ⅳ. In Section Ⅴ, we present the various results of the conducted experiment. Finally, Section Ⅵ gives conclusion and discussion.

Ⅱ. Related Work

Initially, TCP was designed for wired links so it is based on some assumption related to specific characteristic of wired links. These conventional TCP protocols are inadequate for satellite communication systems. Because their congestion control algorithm assumes any loss to be caused by congestion. Comparative analysis of several techniques to improve the E2E performance of TCP over lossy wireless links have been studied[11].

Many researchers have been dedicated to improve TCP performance in satellite networks. Standard TCP often fails to fully utilize the network capacity in large links with large bandwidth-delay product due to the limitation in its conservative congestion control algorithm. Much work has been done and is ongoing to improve the connection's throughput by adopting more aggressive loss-based congestion control algorithms[1]. They typically involve tuning TCP so that the long RTTs representative of satellite links do not negatively impact performance. Two versions that perform well over satellite networks are TCP CUBIC[4] and TCP Hybla[5]. The congestion of window of these TCP protocols increase independent of the propagation delay. CTCP is developed by Microsoft, which is designed to provide good bandwidth scalability with improved RTT fairness, and good TCP-fairness, irrelevant to the windows size[6, 12]. It combines the loss-based approach with the delay-based approach due to which its flows are able to get their fair share of throughput in a mixed environment. In general, these enhanced high-speed TCP protocols achieve high throughput even in long propagation delay (RTT: 520 ms). However, they decrease their performance aggressively as the transmission error increases.

Other TCP protocols like ASTP[13] have been designed to adapt satellite characteristics and attain throughput levels very close to the available satellite capacity. Since standard TCP is globally deployed in all end systems, the effectiveness of these solutions is limited by the fact that they are not universally supported by all end systems over the Internet, due to the specialized nature of the protocol and issues related to fairness with other TCP variants.

Ⅲ. System Model

The system architecture considered in this work is shown in Fig. 1. It consists of gateway, satellite, satellite terminal, Internet, server and client. The gateway, sometimes called as ground station, provides interface between satellite system and terrestrial networks or service providers, which is reaching a satellite terminal through a satellite. All the data requested by the clients pass through the gateway. The satellite terminal is the satellite receiver that is connected to the clients through terrestrial networks or Internet. The satellite is bent-pipe GEO-repeater, which it is only reflect or repeat received traffic from transmitter to receiver on the ground. The satellite segment (gateway to terminal or vice versa) have a large propagation delay (average GEO bi-directional RTT: 520ms).

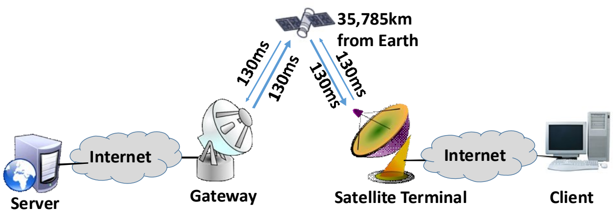

그림 1. 시스템 구조
Fig 1. System architecture.

Proxy is placed at the edges of a satellite link so that all traffic to traverse the satellite link will go through it with the purpose of transparently splitting the E2E TCP connection between two end hosts into three separate connections (distributed) as shown in Fig. 2. Two proxies exist between the end hosts, and they intercept the data on each end host's side of the satellite link. The three connections are established between the two proxies, server and client. The transport layer protocol for middle segment will be either optimized TCP or another.

The protocol stack architecture of distributed TCP splitting based PEP solution considered in this work is shown in Fig. 3. It divides the E2E connection in to three segments, by isolating the satellite link from the terrestrial network. The terrestrial segments have short RTT and they are also error-free. The satellite

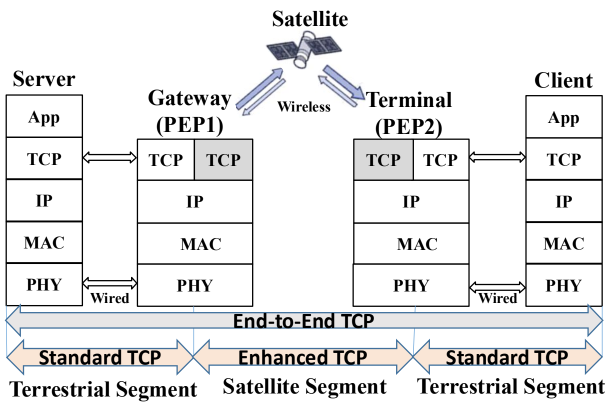

그림 3. 분산 PEP 구조
Fig. 3. Distributed PEP architecture.

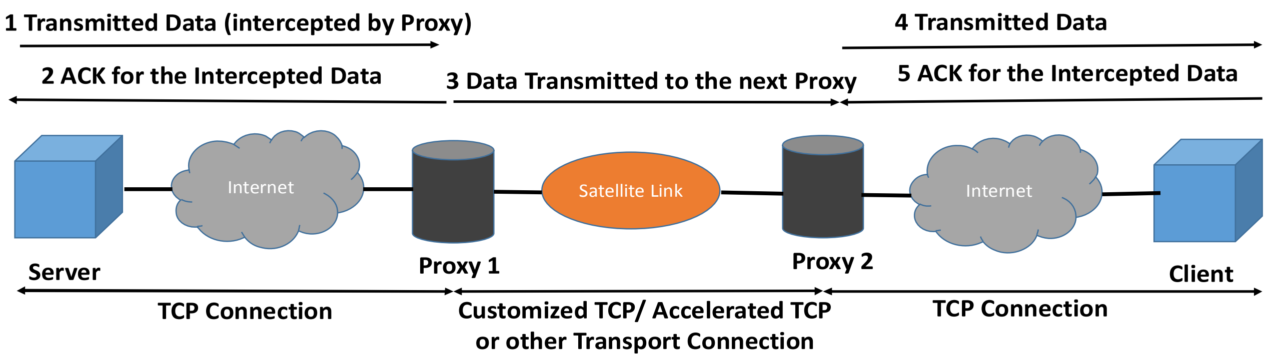

그림 2. 프록시 구조
Fig. 2. Proxy architecture.

segment is characterized by long RTT and error-prone. Therefore, this link is optimized independent of the terrestrial connection.

In the system without PEP solution, the native terminals (client and server) use TCP protocols that have been developed as extensions to the standard TCP version. In enhanced PEP, a TCP flow is generally terminated at the gateway to the satellite link, and a new TCP session is setup on the other side of the satellite link to complete the connection. The gateway receives packets and pretends to be the other side of that connection, and initiates a new connection to the satellite terminal. Here it requires queue handler before direct copy of data between the two sockets, to perform the mechanics of passing packets to and from user space. The second TCP tries to connect with satellite terminal over error-prone wireless channel. In the use of transparent PEP, it requires no modifications to the end nodes. Therefore, the terrestrial segments use conventional (default) TCP deployed at client and server. The satellite segment uses customized TCP like TCP Hybla. In TCP splitting, it appears for the sender that RTT is reduced compared with the standard TCP.

Ⅳ. Testbed Description

The PEP test-bed architecture which is implemented for our experiment is shown in Fig. 4. It contains five boxes to test PEP and E2E TCP variants in satellite communications. PEPsal[14] is installed on both the ends of the satellite segment

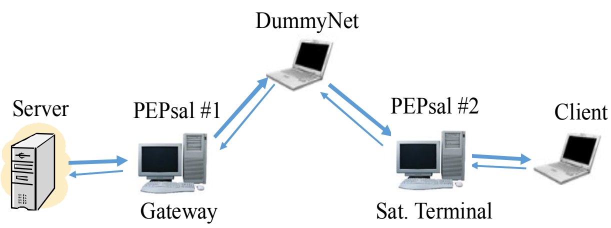

그림 4. PEP 테스트베드 구조
Fig. 4. PEP testbed architecture.

which is open source software for Linux OS. Its aim is to split the connection into segments, and to enhance when sending data to the receiver, and it largely improves TCP performances and friendliness between receivers with different link latencies[10]. It also uses Netfilter[15] to intercept incoming segments and copies them into a queue and the queuer annotates the information (IP addresses and TCP ports) on the two endpoints (here client and server).

The satellite segment is emulated using a network emulating tool, Dummynet[16] that was originally designed for testing network protocols. It enforces queue and bandwidth limitations, delays, packet losses, and multipath effects. It also implements various scheduling algorithms. Dummynet can be used on the machine running the user's application, or on external boxes acting as routers or bridges. It can run in different operating systems, but in our case it runs on FreeBSD. To measure network performance in our testbed, we install iperf[17] in both the client and the server. It is a modern alternative tool written in C language for network performance measurement. It measures the throughput of a network that is carrying them.

Ⅴ. Performance Evaluation

This section presents the results from the conducted experiment on the implemented test scenario. It considers variety of different environments. TCP variants, TCP Reno, CUBIC and Hybla are installed on Linux kernels. We configure the variants on server and client for E2E performance evaluation. In addition, we evaluate CTCP as E2E TCP on client and server windows OS. We compare PEP performance with E2E TCP protocols in different packet loss rates(PER). Moreover, we make TCP performance experiment in presence of multiple connections.

As TCP splitting is transparent to the end users, we use TCP Reno even though CUBIC is default on

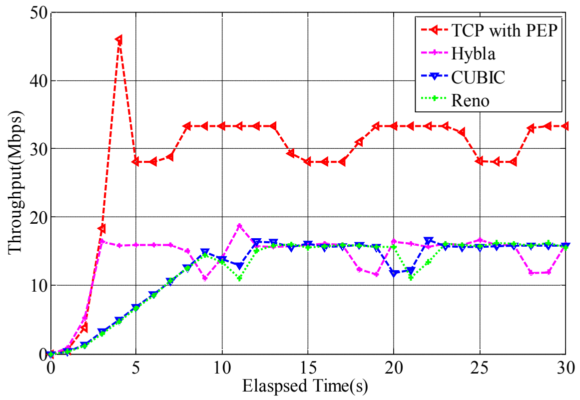

그림 5. BER =0에서 PEP vs TCP Varients의 처리량
Fig. 5. Throughput of PEP vs TCP Varients at BER = 0.

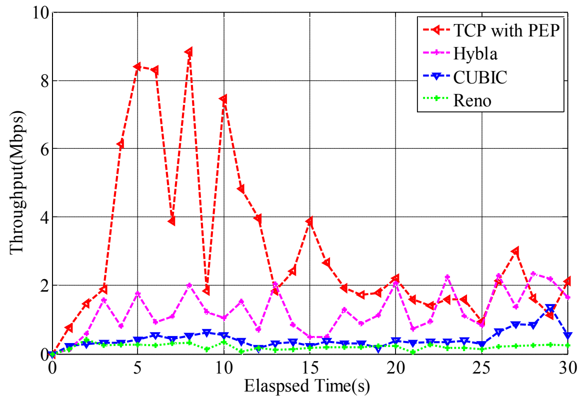

그림 6. BER $=10^{-6}$에서 PEP vs TCP Varients의 처리량
Fig. 6. Throughput of PEP vs TCP Variants at BER= 10^{-6}.

these nodes. To improve the performance, we set Hybla on PEPSal nodes. With this settings, we can have the benefits of TCP/splitting in both directions, from the server to the receiver (forward) and from the client to the server (reverse).

First, we reveal the throughput of PEP and three TCP variants in quasi-error-free condition channel claimed in [18]($BER<10^{-10}$) with 260ms propagation delay in elapsed time of 30 seconds. As it shows, PEP outperforms the other TCP variants. This is shown in Fig. 5. Next, we examine the throughput for the three TCP variants and PEPsal with loss rate. Fig. 6 displays the average throughput (Mbps) of the three TCP variants and connection splitting based TCP solution when BER is 10^{-6}. In this BER, throughput of TCP protocols degrades dramatically, averagely less than 2Mbps. Note that even though the connection splitting based PEP solution achieves better performance than the TCP variants, it decreases its throughput in presence of high error rates.

Standard TCP protocols can not distinguish the origin of packet losses. Link errors cause spurious interference on the congestion control mechanism. This behavior is apparent by examining the performance of TCP variants at different packet error rate (PER) values as shown in Table 1. By contrast, the benefit provided by PEP is entirely to be ascribed to the adoption of TCP Hybla in the satellite segment, which is much more efficient of TCP standard in reopening the congestion window after the spurious reductions caused by random losses. However, the higher PER induces a general performance worsening for all the protocols including the PEP.

Fig. 7 shows throughput of a satellite based connection (single connection) between client and server with respect to BER at RTT of 520ms. PEP with TCP splitting has highest throughput even though decreases its performance as satellite channel becoming more erratic. When comparing the high speed TCP protocols, CTCP is less aggressive and has lower throughput because it uses both loss-based and delay-based algorithms.

In multiple connections, we assume that the

표 1. PEP와 E2E TCP 처리율(RTT = 520ms).
Table 1. PEP and E2E TCP throughput (RTT = 520ms).

TCP Version	TCP Throughput (Mbps)			
	PER = 0%	= 0.1%	= 1%	=10%
Reno	13.969	0.3370	0.0989	0.0045
Cubic	22.197	0.9720	0.2520	0.0132
CTCP	0.6930	0.4920	0.1580	0.0378
PEP	22.300	5.3100	1.5000	0.0559

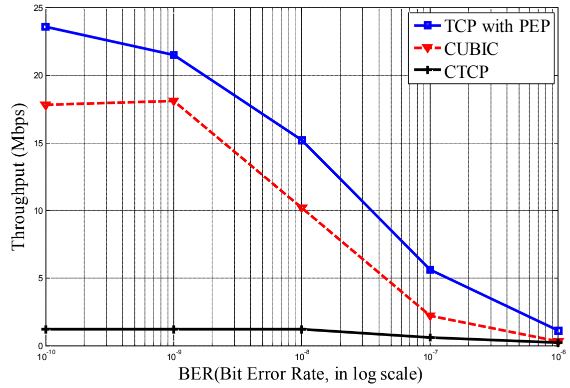

그림 7. BER에 대한 TCP 성능
(RTT = 520ms, BW = 1GB)
Fig. 7. Performance of TCP with respect to BER in single connection(at RTT = 520ms, BW = 1 GB).

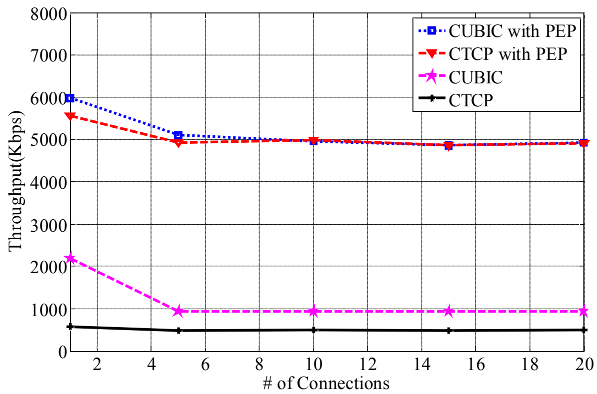

그림 8. 다중 TCP 연결의 처리율
(RTT = 520ms, BER = 10^{-7}, BW = 1GB)
Fig. 8. Throughput of multiple TCP connections.
(at RTT = 520ms, BER = 10^{-7},BW = 1GB)

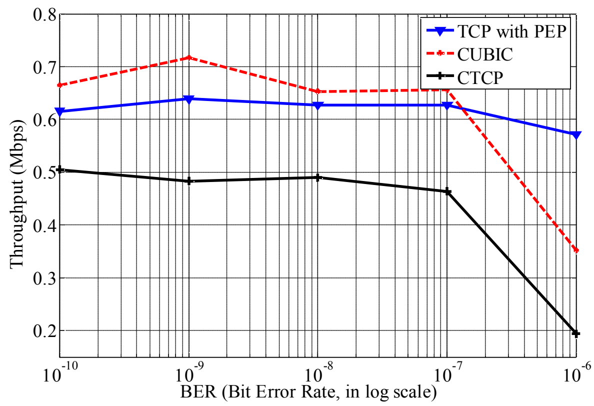

그림 9. 4개의 TCP 연결에서의 TCP 성능
(RTT = 520ms, BW = 4 MB)
Fig. 9. Performance of TCP in 4 TCP connections.
(at RTT = 520ms, BW = 4 MB)

channel bandwidth is 1Gbps and set the BER and RTT of the satellite link as 10^{-7} and 520ms respectively. Fig. 8 illustrates the experimental result of multiple connections with and without PEP, TCP protocol in both Linux and windows operating systems. The result shows that the PEP can improve the throughput significantly over satellite network in simultaneous connections.

In our last experiment, we limit the bandwidth of the satellite channel with 4Mbps and open four simultaneous connections to evaluate the performance with respective to the BER of the satellite channel. Fig. 9 shows the result of this experiment. It is shown that PEP can not improve the TCP throughput as compared to CUBIC because there is packet loss caused by both the congestion and the channel error in environment with the limited bandwidth.

Ⅵ. Conclusions and Discussion

Connection splitting based PEP is a well-known solution for heterogeneous networks that contain satellite segments. Moreover, it does not require to modify the end user's terminal to deploy new TCP protocols. We implement a testbed to conduct experiment to evaluate PEP performance at satellite's RTT and different channel errors.

PEP improves TCP throughput in both single connection and multiple connections in all test scenarios. However, in erratic channel, the performance of the PEP solution decreases rapidly even though it has better throughput compared with E2E TCP variants at same channel error. Therefore, connection splitting based PEP solution is not enough to achieve desirable throughput in erasure satellite channel. It requires to make the satellite link robust and exploit the channel resources in high packet loss rate. In addition, there are still some problems and weaknesses that could affect the quality of PEP performance. The E2E paradigm of TCP is essentially broken by splitting connections. Then, keeping existing security mechanisms in place is often impossible when using TCP connection splitting. Since IPSec prevents the use of PEP, it is required to introduce new security requirements for the proxy.

REFERENCES

[1] A. Pirovano and F. Garcia, "A New survey on improving TCP performances over Geostationary Satellite Link," Network and Communication Technologies, vol. 2, no. 1, Jan. 2013.

[2] Kyu-Hwan Lee, Hyun-Ki Jung, Jae-Hyun Kim, " A Study on PEP based on Cross-layer and Network Coding in DVB-RCS Networks," Journal of the Institute of Electronics and Information Engineers,, vol. 52, no. 3, pp.25-31, March 2015.

[3] C. Caini, R. Firrincieli, M. Marchese, T. d. Cola, M. Luglio, C. Roseti,N. Celandroni, and F. Potorti, "Transport Layer Protocols and Architectures for Satellite Networks," International Journal of Satellite Communications and Networking, vol. 25, no. 1, 2007, pp. 1 - 26.

[4] Ha, I. Rhee, and L. Xu, "CUBIC: A New TCP-Friendly High-Speed TCP Variant," SIGOPS Oper. Syst. Rev., vol. 42, no. 5, pp.64 - 74.Jul. 2008.

[5] C. Caini and R. Firrincieli, "TCP Hybla: A TCP Enhancement for Heterogeneous Networks," International Journal of Satellite Communications and Networking, vol. 22, no. 5, 2004.

[6] Tan. Kun; Jingmin Song; Qian Zhang; Sridharan, M., "A Compound TCP Approach for High-Speed and Long Distance Networks," INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings, pp.1, 12, April 2006.

[7] Comparative Performance Evaluation of TCP Hybla and TCP Cubic for Satellite Communication Under Low Error Conditions

[8] Fei Peng; Cardona, A.S.; Shafiee, K.; Leung, V.C.M., "TCP Performance Evaluation over GEO and LEO Satellite Links between Performance Enhancement Proxies," Vehicular Technology Conference (VTC Fall), 2012 IEEE , vol., no., pp.1,5, 3-6 Sept. 2012

[9] P. Davern, N. Nashid, C. J. Sreenan, and A. Zahran, "HTTPEP: a HTTP Performance Enhancing Proxy for Satellite Systems," International Journal of Next Generation Computing (IJNGC), vol. 2, 2011.

[10] C. Caini.; Firrincieli, R.; Lacamera, D., "PEPsal: a Performance Enhancing Proxy designed for TCP satellite connections" Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd , vol.6, pp.2607,2611, 7-10 May 2006

[11] Maisuria, J.V.; Patel, R.M., "Overview of Techniques for Improving QoS of TCP over Wireless Links," Communication Systems and Network Technologies (CSNT), 2012 International Conference on , vol., no., pp.366,370, 11-13 May 2012

[12] Tan, J. Song, Q. Zhang, and M. Sridharan, "A Compound TCP Approach for High-Speed and Long Distance Networks," in Proc. IEEE INFOCOM, 2006.

[13] M. Muhammad, F. Kasmis, and T. de Cola, "Advanced Transport Satellite Protocol," in Global Communications Conference (GLOBECOM), 2012 IEEE, Anaheim, CA, USA, 2012, pp. 3299 - 3304.

[14] PEPsal source code, available at: http://www.sourceforge.net/projects/pepsal/ [9]S. Poojary, and V. Sharma, "Theoretical analysis of high-speed multiple TCP connections through multiple routers," in Proc. IEEE ICC, 2013

[15] The Netfilter/iptables project, firewalling, NAT and packet mangling for Linux. http://www.netfilter.org.

[16] DummyNet, available at: http://info.iet.unipi.it/~luigi/dummynet/

[17] Iperf source software, available at: http://software.es.net/iperf/

[18] ETSI. Digital video broadcasting (DVB); interaction channel for satellite distribution systems. Final draft EN 301 790, V1.3.1 (2002-11). ETSI, 2002.

저 자 소 개

네 띠 넬(학생회원)
2008년 Mekelle Institute of Technology Electronics and Communication Engineering 졸업
2013년～현재 Ajou University Electrical and Computer Engineeringin Master Degree
<주관심분야 : TCP performance in satellite communication, Performance enhancing proxy, Cross-layer, Wireless communication등>

김 종 무(학생회원)
2015년 연세대학교 컴퓨터공학과 졸업
2015년～현재 아주대학교 전자공학과 석사과정
<주관심분야 : 위성 통신, WAN/위성통신 가속기, Performance enhancing proxy, 등>

이 규 환(학생회원)
2007년 아주대학교 전자공학부 졸업
2007년～현재 아주대학교 전자공학과 석/박사 통합과정
<주관심분야 : 네트워크 코딩, 위성통신, WLAN, 무선망 QoS, Ad-hoc, Mesh network 등>

김 재 현(평생회원)
1987년～1996년 한양대학교 전산과 학사 및 석/박사 졸업
1997년～1998년 미국 UCLA 전기전자과 박사 후 연수
1998년～2003년 Bell Labs, NJ, USA, 연구원
2003년～현재 아주대학교 전자공학부 교수
<주관심분야 : QoE/QoS, 무선 MAC 프로토콜, IEEE 802.11/15, B5G 통신 시스템, 국방 전술네트워크, 위성시스템 등>