

위성 통신에서 Cross-layer 기반 PEP 성능 평가

김 중 무*, Nathnael Gebregziabher W.*, 이 규 환*, 김 재 현^o

Performance Evaluation of PEP Based on Cross-Layer in Satellite Communication System

Jong-Mu Kim*, Nathnael Gebregziabher W.*, Kyu-Hyan Lee*, Jae-Hyun Kim^o

요 약

위성 통신은 넓은 지역에 통신 서비스를 제공하는 광역 네트워크이다. 하지만, 위성 통신은 대역폭이 제한되고 매우 긴 전파 지연 시간과 높은 BER(Bit Error Rate)로 인해 위성 통신을 고려하지 않은 TCP의 성능이 저하 된다. 본 논문에서는 위성 통신에서의 TCP 성능을 향상시키기 위해 DVB-RCS(Digital Video Broadcasting-Return Channel via Satellite) 네트워크 환경을 고려한 cross-layer 기반 PEP(Performance Enhancing Proxy) 기술을 제안 한다. 제안하는 프로토콜은 TCP와 링크 계층 간 위성 자원 할당 정보를 정보 교환을 통해 최적의 TCP CWND(Congestion Window)를 설정한다. 제안한 프로토콜의 성능 평가를 위해 리눅스 기반의 PEP 테스트 베드 를 구현하였다. 성능 평가 결과 다양한 BER에서 제안한 프로토콜은 위성 자원 할당 정보를 이용하여 최적의 TCP CWND 크기를 설정하기 때문에 단일 및 다중 세션 환경에서 기존 TCP 보다 제안한 프로토콜이 더 좋은 성능을 보여준다는 것을 확인할 수 있었다.

Key Words : PEP, Cross-layer, Congestion Window, TCP splitting, DVB-RCS, BER

ABSTRACT

Satellite communication is a wide area network (WAN) which provides communication service worldwide. However, the performance of TCP can be seriously degraded in the satellite networks due to limited bandwidth, long round-trip time (RTT) and high bit error rate (BER) over satellite links. In order to improve the performance of TCP, this paper proposes cross-layer Performance Enhancing Proxy (PEP) in digital video broadcasting-return channel via satellite (DVB-RCS) networks. The proposed protocol sets TCP Congestion Window (CWND) size by using satellite resource allocation information exchanged between TCP and the link-layer. we implement PEP testbed based on Linux to evaluate the performance of the proposed protocol. The simulation results show that the proposed protocol performs better than standard TCP both in single and multiple sessions in variant BER, because the proposed protocol sets TCP CWND size by using satellite resource allocation.

※ 본 연구는 방위사업청과 국방과학연구소가 지원하는 국방위성항법특화연구센터 사업의 일환으로 수행되었습니다.

♦ First Author : Ajou University Department of Electrical and Computer Engineering, blackkim822@ajou.ac.kr, 학생회원

o Corresponding Author : Ajou University Department of Electrical and Computer Engineering, jkim822@ajou.ac.kr, 종신회원

* Ajou University Department of Electrical and Computer Engineering, natumit2000@gmail, lovejiyeon7@ajou.ac.kr, 학생회원

논문번호 : KICS2015-10-349, Received October 30, 2015; Revised January 11, 2016; Accepted January 13, 2016

I. 서 론

위성 통신은 넓은 지역에 통신 서비스를 제공 하는 광역 네트워크(WAN)이다. 따라서 지상망을 이용하여 통신하기 힘든 산악지역, 선박 및 항공기 통신에 매우 유용하게 사용되고 있다^[1]. 하지만, 위성 통신에서 TCP를 이용한 통신을 수행하면 다음과 같은 3가지 원인에 의해 성능이 현저하게 저하 될 수 있다^[2]. 첫 번째로 매우 많은 사용자가 제한된 대역폭을 공유하기 때문에 사용자당 대역폭이 제한된다. 두 번째로 위성 통신은 매우 긴 RTT(Round Trip Time)를 나타내는데, 긴 RTT는 TCP CWND(Congestion Window) 크기의 상승 속도를 낮추어 최대 전송 속도에 도달하는 시간을 늦추기 때문에 지상망보다 동일 대역폭 대비 평균 전송 속도가 낮다^[3]. 마지막으로 위성망은 지상망에 비해 높은 BER(Bit Error Rate)을 보이는데, TCP는 비트 에러를 혼잡 발생 상황으로 오인하여 불필요한 혼잡 제어 알고리즘을 수행하게 하므로 이는 또 다른 전송 속도를 낮추는 원인이 된다.

위성 통신에서의 TCP 성능 저하를 개선하기 위한 방법 중 하나로 PEP(Performance Enhancing Proxy)에 대해 많은 연구들이 수행되었다^[4-6]. PEP는 네트워크에 설치되어 종단 간 통신 성능을 향상시키기 위해 사용되는데, 일반적으로 많이 사용되는 TCP 연결 분할 방식 PEP는 종단점을 지상망 구간과 위성망 구간으로 분리하고, 위성망 구간에서는 위성 링크에 최적화된 파라미터를 적용하여 위성 통신에서의 TCP 성능을 향상시킨다. 하지만, 기존 TCP 연결 분할 방식 PEP는 BER이 높은 환경에서는 TCP의 성능이 여전히 대역폭 대비 낮은 성능을 보인다^[7].

이러한 문제점을 해결하기 위해 본 논문에서는 DVB-RCS(Digital Video Broadcasting-Return Channel via Satellite) 네트워크에서 cross-layer 기술을 기반으로 한 TCP 연결 분할 방식 PEP를 제안하고 이에 대한 성능 평가를 수행한다. 제안하는 프로토콜에서는 cross-layer 기법을 활용하여 링크 계층에서 획득한 자원 할당 정보를 TCP에 전달하고, TCP는 전달받은 자원 할당 정보를 활용하여 최적의 TCP CWND 크기를 설정하여 TCP의 성능을 향상 시킨다. 제안하는 프로토콜의 성능 평가를 위해 테스트베드를 구축하고, 리눅스 오픈 소스인 PEPsal과 위성 세그먼트를 설정하기 위해 Free BSD의 Dummynet을 사용하여 성능 평가를 수행 한다.

본 논문의 구성은 다음과 같다. II장에서는 관련연구를 살펴보고 III장에서는 제안하는 프로토콜을 기술

한다. IV장에서는 제안하는 프로토콜을 테스트베드를 이용하여 성능을 평가하고, 마지막 V장에서 결론을 맺는다.

II. 관련 연구

2.1 TCP Variants

TCP는 1981년 IETF(Internet Engineering Task Force)를 통해 발표된 RFC 793에서 공식적으로 정의된 프로토콜이며 꾸준히 개선되어 TCP Reno등과 같은 변형 프로토콜들이 정의되었다. 하지만, TCP는 유선 링크에서의 오류가 매우 적은 환경을 기반으로 디자인되었기 때문에 위성 통신 시스템에는 적합하지 않다. 특히 TCP의 혼잡 제어 알고리즘은 비트 에러로 인한 세그먼트 손실도 TCP 혼잡 발생 상황으로 인지하여 TCP 혼잡 제어 알고리즘을 발생시키기 때문에 위성 통신에서 TCP 사용 시 성능이 저하된다. 이러한 위성 통신 환경에서의 성능을 향상시키기 위해 제안된 대표적인 TCP에는 TCP CUBIC과 TCP Hybla가 있다. 본 절에서는 주요 TCP 프로토콜들을 간략하게 소개한다.

2.1.1 TCP Reno

TCP Reno는 유선 링크에서 사용되는 대표적인 TCP로, ACK(Acknowledge) 패킷을 기반으로 TCP CWND 크기를 조절하는 느린 시작(slow start), 중복된 ACK 패킷을 3개 이상 받으면 패킷 손실이 발생했다고 판단하여 즉시 해당 패킷을 재전송하는 빠른 재전송(fast retransmit)과 빠른 회복 알고리즘(fast recovery)을 적용한 TCP 프로토콜이다^[8]. 그러나 이는 유선 링크만 고려하였기 때문에 위성 통신에서 성능이 좋지 않다.

2.1.2 TCP CUBIC

TCP CUBIC은 고속 네트워크에서 좋은 성능을 내기 위해 고안된 BIC TCP(Binary Increase Congestion control)를 향상시킨 TCP 프로토콜이다^[9]. 삼차 함수를 활용하여 TCP CWND 크기를 빠르게 증가시켜 지상망과 같은 고속 네트워크에서 높은 전송률을 보이지만, 추가적인 대역폭을 쓰기 위해 TCP CWND 크기를 급격하게 증가시켜 다량의 패킷 손실이 발생하는 문제가 있다.

2.1.3 TCP Hybla

TCP Hybla는 위성 통신과 같은 긴 RTT를 보유한

네트워크 환경에서의 TCP 성능 저하를 극복하기 위해 제안된 프로토콜이다¹⁰⁾. 위성망은 매우 긴 RTT를 나타내므로 이를 고려하지 않은 TCP를 사용했을 때 TCP가 최대전송률에 도달하는 시간이 늦어지며 이는 평균 전송 속도를 낮추어 심각한 성능 저하 현상이 나타난다. 이러한 문제를 해결하기 위해 TCP Hybla는 실제 RTT가 아닌 지상망과 같이 낮은 RTT를 가진 가상 링크를 참조하여 위성망을 활용하여도 지상망과 비슷한 전송률이 나오도록 조절한다.

2.2 DVB-RCS

DVB-RCS는 양방향 위성통신 시스템 표준으로, 위성 단말에서 위성 기지국으로 데이터를 전송하는 채널인 역방향 채널에 대한 상세 표준이다¹¹⁾. DVB-RCS를 사용하는 위성 통신 시스템은 그림 1과 같이 위성 단말(Satellite Terminal), 게이트웨이(Gateway), NCC(Network Control Center)로 구성되어 있다. 순방향 링크에서는 TDM(Time Division Multiplexing) 방식을 사용하고, 역방향 링크에서는 DAMA(Demand Assignment Multiple Access) 프로토콜을 이용하여 데이터를 전송한다. NCC는 위성 단말들이 요청한 자원을 순방향 채널의 TBTP(Terminal Burst Time Plan)에 자원 할당 정보를 포함하여 위성 단말들에게 알려 준다. DVB-RCS 표준은 다음과 같은 고정된 채널 할당인 CRA(Continuous Rate Assignment)와 동적 할당인 RBDC(Rate-Based Dynamic Capacity)와 VBDC (Volume-Based Dynamic Capacity)를 제공한다¹²⁾. CRA는 위성 단말 초기화 시 위성 NCC에게 고정된 채널을 할당받고, RBDC는 위성 단말이 필요한 데이터 전송률을 측정하여 그에 따라 위성 NCC에게 위성 자원을 요청하여 할당 받는다. 마지막으로 VBDC는 위성 단말이 버퍼에 쌓여있는 데이터를 기반으로 위성 자원을 요청하여 할당 받는다.

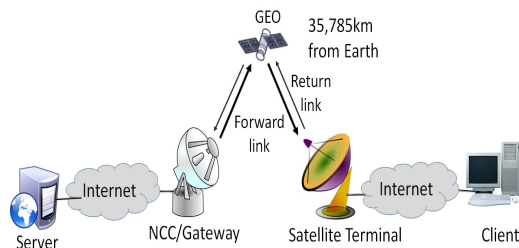


그림 1. DVB-RCS 구조
Fig. 1. DVB-RCS architecture

2.3 Cross-layer 기반 PEP 연구 동향

PEP에서는 물리 및 링크 계층과 전송 계층 사이에서 버퍼 크기나 자원 할당 등의 정보를 교환하여 매우 효과적으로 TCP 전송 성능을 향상 시킬 수 있다. Peng의 연구에서는 TCP와 링크계층간의 정보 교환을 통하여 대기열의 오버플로우를 방지하는 TCP 연결 분할 방식을 제안했다¹³⁾. Luglio와 Alins의 연구에서는 TCP의 전송 속도를 하위 계층의 버퍼 크기를 고려하여 조절하는 기법이 제안되었다^{14,15)}. Park의 연구에서는 TCP에서 물리계층의 ACM(Adaptive Coding & Modulation)에 대한 정보를 활용하는 기법을 제안했다¹⁶⁾. Lee의 연구에서는 링크계층의 자원 할당 정보를 활용하여 TCP 전송 속도를 정확히 조절 하는 기법을 제안했다¹⁷⁾. 하지만, Lee의 연구에서는 실제로 테스트베드를 활용한 성능 평가는 하지 않았다. 따라서 본 논문에서는 cross-layer 기반 PEP를 제안하고 성능 평가를 위한 테스트베드를 구축하고 성능평가를 한다.

III. 제안하는 프로토콜

3.1 시스템 모델

본 논문에서는 그림 2와 같은 구조의 cross-layer 기반 TCP 연결 분할 방식 PEP 네트워크 구조를 제안한다. 제안하는 프로토콜의 시스템 모델을 살펴보면, 서버와 클라이언트 사이에 위성 통신을 위한 게이트웨이와 위성 단말이 있다. 게이트웨이와 위성 단말에는 PEP가 탑재되어 있고, 서버와 게이트웨이, 위성 단말과 클라이언트는 유선으로 연결 되어 있으며, 게이트웨이와 위성 단말은 위성 링크로 연결되어 있다. 성능을 향상시키기 위해 위성 링크와 지상망의 유선 링크를 분할하여 지상망에는 지상망에 적합하게 설계된 기존의 standard TCP를 사용하고 위성 링크에는 본 연구에서 제안한 프로토콜이 적용된다. 위성 단말의

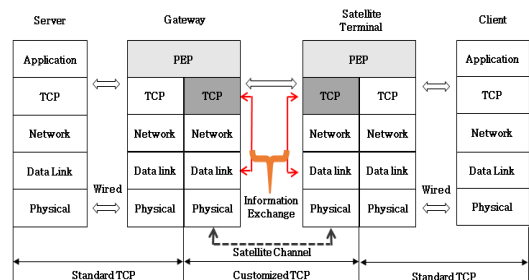


그림 2. 제안하는 프로토콜의 시스템 모델
Fig. 2. System model of the proposed protocol

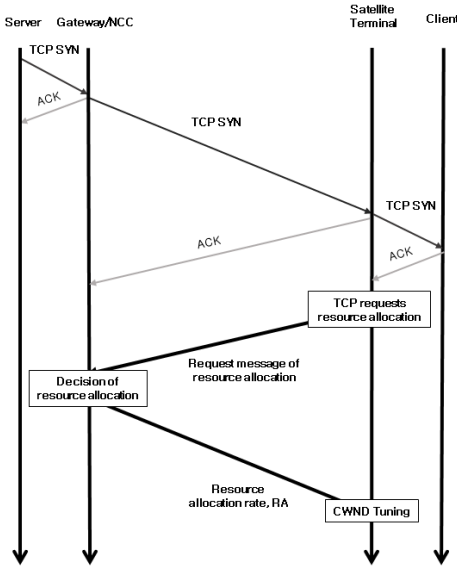


그림 3. TCP 연결 초기화 및 cross-layer 정보 교환
Fig. 3. Initial TCP connection and procedure of cross-layer information exchange

링크 계층에서는 RBDC를 이용하여 자원할당을 받는다고 가정한다. RBDC를 이용하므로 위성 단말은 필요한 데이터 전송률과 이를 바탕으로 NCC로부터 할당받은 위성 링크 자원량을 알 수 있다.

제안한 기법에서는 그림 3에서 보는 것과 같이 TCP 연결 초기화시, 게이트웨이의 PEP는 서버에서 클라이언트로 보내는 TCP SYN(Synchronous idle) 메시지를 가로채어, TCP 연결 분할을 수행한다. 게이트웨이는 가로챈 메시지를 저장하고 클라이언트와의 TCP 연결을 위한 SYN 메시지를 클라이언트에게 보내고 ACK 패킷을 서버에게 보낸다. 서버는 ACK 패킷을 받는 즉시 다음 패킷을 전송한다. 위성 단말의 PEP에서도 마찬가지로 게이트웨이의 PEP에서 클라이언트로 보내는 TCP SYN 메시지를 가로채어 TCP 연결 분할을 발생시킨다. 위성 단말은 ACK 패킷을 게이트웨이에게 보내고 가로챈 TCP SYN 메시지를 클라이언트에 전송하고 클라이언트가 보낼 ACK 패킷을 기다린다. TCP SYN 메시지를 받은 클라이언트는 위성 단말에게 ACK 패킷을 보낸다.

3.2 제안하는 CWND 기법

기존의 위성 링크에 TCP Hybla를 사용한 연결 분할 방식 PEP는 위성 링크의 자원 할당 정보를 알지 못한다. 따라서, 최적의 CWND 크기를 설정하는데 시간이 필요하다. 또한 비트 에러를 혼잡 발생 상황으

로 오인하여 불필요한 혼잡 제어 알고리즘을 수행으로 인해 CWND 크기를 감소 시켜, BER이 높은 환경에서 대역폭 대비 낮은 성능을 보인다. 이러한 문제를 해결하기 위해, 제안하는 프로토콜은 cross-layer 기법을 활용하여 TCP와 링크 계층 간 정보를 교환한다. 위성 링크를 통해 최적의 TCP CWND 크기를 설정하기 위해, 그림 3에서 보는 것과 같이 위성 단말에서 TCP는 링크 계층에게 필요한 자원 할당 정보를 요청하고, 링크 계층은 NCC에게 자원 할당 정보를 요청한다. NCC는 채널 상태와 위성 단말의 수 등을 고려하여 자원 할당 정보를 링크 계층에 알려준다. 링크 계층은 전달 받은 자원 할당 정보를 TCP에게 알려준다. 위성 단말에서는 수신한 자원할당 정보를 이용하여 TCP CWND 크기를 조절한다. TCP CWND 크기는 식 (1)과 같이 계산한다. RTT 는 round trip time, R_A 는 자원 할당 값, K 는 프로토콜 오버헤드 비율, $TCPsgmsize$ 는 TCP segment size이다.

$$CWND = \frac{RTT * R_A}{(1 + K) * TCPsgmsize} \quad (1)$$

IV. 성능 평가

본 논문에서 제안한 프로토콜의 성능을 평가하기 위해 그림 4와 같이 PEP 테스트베드를 구축하였다. 테스트베드 환경은 표 1과 같다. OS는 LINUX (Kernel 3.17.4)를 설치하였고, TCP 성능 검증을 위해 Client와 Sever에 Iperf(Version 3.0.6)을 설치하였다. Satellite emulator에는 FreeBSD 운영체제를 설치하였으며 위성 환경을 에뮬레이션 하기위해 FreeBSD 운영체제에 기본 내장된 Dummynet을 사용하여 RTT와 BER을 조정하였다^[18,19]. 파라미터의 환경 설정은 표 2와 같다. 위성 환경에 맞게 bandwidth는 8 Mbps, RTT는 500 ms로 설정하였다. 제안하는 프로토콜과 TCP CUBIC, TCP Hybla, TCP Reno, 위성 링크에 TCP Hybla를 사용한 TCP 연결 분할 방식 PEP(TCP-split)의 기본적 성능에 대한 검증을 위해

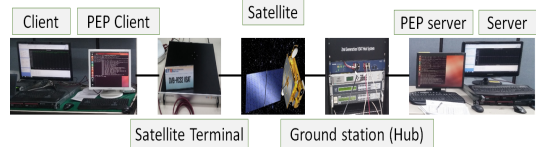


그림 4. PEP 테스트베드 구성도
Fig. 4. PEP testbed architecture

표 1. 테스트베드 환경

Table 1. Testbed environment

Testbed components	Detailed description
Client	OS: Linux (Kernel 3.17.4) 1 giga LAN interface Iperf (Version 3.0.6)
PEPsal	OS: Linux (Kernel 3.17.4) 2 giga LAN interfaces PEPsal (Version 2.0.1)
Satellite emulator	OS: FreeBSD 2 giga LAN interfaces DummyNet
PEPsal	OS: Linux (Kernel 3.17.4) 2 giga LAN interfaces PEPsal (Version 2.0.1)
Server	OS: Linux (Kernel 3.17.4) 1 giga LAN interface Iperf (Version 3.0.6)

표 2. 파라미터 환경 설정

Table 2. Setup configuration parameter

Parameter	Value
Bandwidth (Satellite emulator)	8 Mbps
RTT (Satellite emulator)	500 ms
Maximum number of segments	326
Queue size	500 bytes

표 3. TCP 처리율 (RTT = 0 ms, BER = 0, 1 Mbps)

Table 3. TCP throughput (RTT = 0 ms, BER = 0, 1 Mbps)

TCP version	TCP throughput (Mbps)
TCP Reno	94.1
TCP CUBIC	94.2
TCP HYBLA	94.2
TCP-split	94.2
Proposed	94.2

실험 환경을 RTT는 0 ms, BER은 0으로 설정한 후 단일 TCP 세션의 전송률을 측정 및 비교하였다. 측정 결과는 표 3과 같다. 이러한 실험 환경에서는 전송률이 제한하는 프로토콜과 TCP CUBIC, TCP Hybla, TCP Reno, TCP-split 모두 비슷하게 나오는 것을 볼 수 있다.

그림 5는 비트 에러가 없는 위성 환경에서 제안한 프로토콜을 사용했을 때 TCP CWND 크기의 변화와 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split을 사용했을 때 TCP CWND 크기의 변화를 시간의 변화

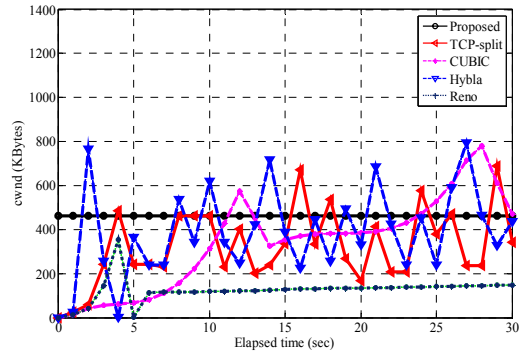


그림 5. 위성통신에서 TCP CWND (BER = 0)

Fig. 5. TCP CWND in satellite networks (BER = 0)

에 따라 비교한 것을 보여준다. 그림 5에서 보는 것과 같이 TCP Reno는 긴 RTT 때문에 최적의 TCP CWND 크기를 찾는데 오랜 시간이 필요함을 확인할 수 있다. TCP Hybla와 TCP CUBIC은 최적의 TCP CWND 크기를 설정하기 위해 자체 알고리즘을 사용하여 최적의 TCP CWND 크기를 설정하려고 시도하나 크기가 심하게 변한다. TCP-split은 위성망과 지상망을 분리하였지만, 위성 망에 TCP Hybla와 비슷하게 CWND 크기를 설정하는 것을 볼 수 있다. 반면, 제안하는 프로토콜은 자원 할당 정보를 이용하므로 최적의 TCP CWND 크기를 초기부터 설정하여 사용하는 것을 볼 수 있다.

그림 6은 비트 에러가 있는 위성 환경에서 제안한 프로토콜을 사용했을 때 TCP CWND 크기의 변화와 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split을 사용했을 때 TCP CWND 크기의 변화를 시간의 변화에 따라 비교한 것을 보여준다. BER은 10^{-6} 으로 가정하였다. 그림 6에서 보는 것과 같이 TCP Reno와 TCP CUBIC은 비트 에러에 심각하게 영향을 받아

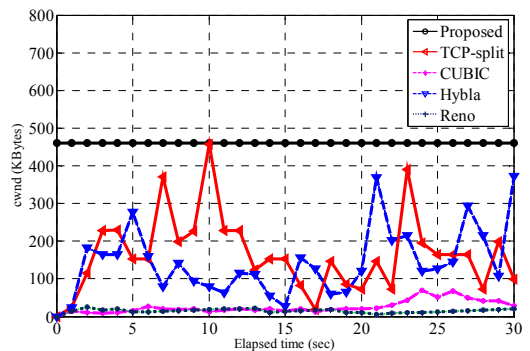


그림 6. 위성통신에서 TCP CWND (BER = 10^{-6})

Fig. 6. TCP CWND in satellite networks (BER = 10^{-6})

TCP CWND 크기를 증가시키지 못하였다. TCP Hybla는 TCP Reno, TCP CUBIC에 비해 TCP CWND 크기를 증가 시켰지만 비트 에러에 많은 영향을 받았다. TCP-split은 CWND 크기를 많이 증가시켰지만, 자원 할당 정보를 알지 못하기 때문에 비트 에러로 인한 불필요한 혼잡 제어 알고리즘으로 인해 CWND 크기가 변하는 것을 볼 수 있다. 반면, 제안하는 프로토콜은 자원 할당 정보를 이용하여 최적의 TCP CWND 크기를 설정하였기 때문에 비트 에러의 영향을 받지 않는 것을 볼 수 있다.

그림 7은 비트 에러가 있는 위성 환경에서 제안한 프로토콜을 사용했을 때 평균 TCP 처리율과 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split을 사용했을 때 평균 TCP 데이터 처리율을 위성 링크에서 BER에 따라 비교한 것을 보여준다. 그림 7에서 보는 것과 같이 제안한 프로토콜이 기존의 TCP들 보다 평균 TCP 처리율이 더 좋은 결과를 나타내는 것을 알 수 있다. BER이 10^{-7} 일 때 제안한 프로토콜은 TCP Reno보다 TCP 처리율이 493% 향상되었다. BER이 10^{-6} 일 때 TCP-split보다 제안한 프로토콜의 TCP 처리율이 199% 향상되었다. 제안한 프로토콜은 위성 자원 할당 정보를 활용하여 최대로 이용 가능한 TCP CWND 크기를 설정하였기 때문에 TCP 처리율이 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split보다 더 높은 것을 확인할 수 있었다.

그림 8은 비트 에러가 있는 위성 환경에서 다중 세션 일 때 제안한 프로토콜을 사용 했을 때 각 세션의 TCP 처리율의 평균값과 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split을 사용 했을 때 각 세션의 TCP 처리율의 평균값을 위성 링크에서 BER에 따라 비교한 것을 보여준다. 세션의 수는 5개로 가정하였

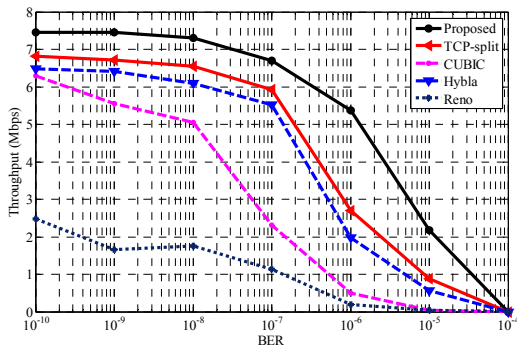


그림 7. 단일 세션일 때 TCP 처리율
Fig. 7. TCP throughput in presence of single connection

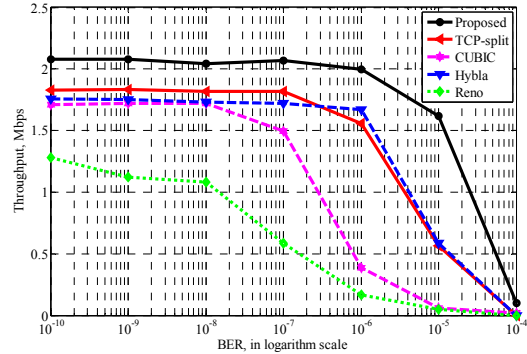


그림 8. 세션이 5개 일 때 TCP 처리율
Fig. 8. TCP throughput in presence of five connections

다. 그림 8에서 보는 것과 같이 BER이 10^{-6} 일 때 제안한 프로토콜의 TCP 처리율의 TCP Reno의 TCP 처리율보다 284% 향상되었다. BER이 10^{-5} 일 때 TCP-split보다 TCP 처리율이 286% 향상 되었다. 다른 연결이 존재하는 상황에서도 제안한 프로토콜이 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split보다 평균 TCP 처리율이 더 높다는 것을 확인 할 수 있었다.

V. 결론

본 논문에서는 위성 통신에서 제한된 대역폭, 긴 왕복시간과 위성 링크에서의 비트 에러로 인해 TCP의 성능 저하를 해결하기 위해 DVB-RCS 네트워크에서 cross-layer 기반 TCP 연결 분할 방식 PEP를 제안하였다. 성능 분석 결과 위성 자원 할당 정보를 활용하여 비트 에러가 있는 환경과 없는 환경 모두 최적의 TCP CWND 크기를 설정하는 것을 확인 할 수 있었고, 단일 연결 및 다중 연결에서 TCP Reno, TCP Hybla, TCP CUBIC, TCP-split에 비해 좋은 TCP 데이터 처리율을 보이는 것을 확인 할 수 있었다. 따라서 본 논문에서 제안하는 프로토콜은 DVB-RCS 네트워크에서 TCP의 성능을 향상시킬 수 있을 것으로 기대된다.

References

[1] M. P. Anastopoulos, T. Taleb, P. G. Cottis, and M. S. Obaidat, "Feedback suppression in multicast satellite networks using game theory," *IEEE Systems J.*, vol. 6, no. 4, pp. 657-666, 2012.

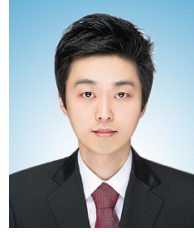
- [2] A. Pirovano and F. Garcia, "A new survey on improving TCP performances over geostationary satellite link," *Network and Commun. Technol.*, vol. 2, no. 1, Jan. 2013.
- [3] M. Allman, C. Hayes, H. Kruse, and S. Osterman, "TCP performance over satellite links," in *Proc. 5th Int. Conf. Telecommun. Syst.*, pp. 456-469, Mar. 1997.
- [4] C. Caini, R. Firrincieli, and D. Lacamera, "PEPsal: A performance enhancing proxy for TCP satellite connections," *IEEE Aerospace and Electronic Syst. Mag.*, vol. 22, no. 8, pp. B-9 - B-16, Aug. 2007.
- [5] N. G. Weldegiorgis, K. H. Lee, Y. J. Choi, and J. H. Kim, "Testbed and discussion for PEP in satellite communications," in *Proc. ICEIC 2015*, pp. 28-31, Singapore, Jan. 2015.
- [6] M. K. Park, D. G. Oh, Y. J. Choi, and J. H. Kim, "Building Three-segment splitting PEPsal testbed and performance evaluation," in *Proc. KICS Winter Conf.*, pp. 998-999, Jan. 2015.
- [7] M. Luglio, C. Roseti, and F. Zampognaro, "Performance evaluation of TCP-based applications over DVB-RCS DAMA schemes," *Int. J. Satellite Commun. Netw.*, vol. 27, no. 3, pp. 163-191, 2009.
- [8] J. F. Kurose, *Computer Networking: A Top-Down Approach Featurin g the Internet*, 3Ed., Pearson Education India, 2005.
- [9] S. Ha, I. Rhee, and L. Xu, "CUBIC: a new TCP-friendly high-speed TCP variant," *ACM SIGOPS Operating Syst. Rev.*, vol. 42, pp. 64-74, Jul. 2008.
- [10] C. Caini and R. Firrincieli. "TCP Hybla: a TCP enhancement for heterogeneous networks," *Int. J. Satellite Commun. Netw.*, vol. 22, no. 5, pp. 5547-566, 2004.
- [11] "Digital Video Broadcasting(DVB); Second Generation DVB Interactive Satellite System (DVB-RCS2); Part 2: Lower Layers for Satellite standard," ETSI EN 301 545-2, pp. 1-195, 2012.
- [12] D. K. Petraki, M. P. Anastasopoulos, and P. G. Cottis, "Dynamic resource allocation for DVB-RCS networks," *Int. J. Satellite Commun. Netw.*, vol. 26, no. 3, pp. 189-210, May/Jun. 2008.
- [13] F. Peng, L. Wu, and V. C. M. Leung, "Cross-layer enhancement of TCP split-connections over Satellites Links," *Int. J. Satellite Commun. Netw.*, vol. 24, no. 5, pp. 405-418, 2006.
- [14] M. Luglio, F. Zampognaro, T. Morell, and F. Vieira, "Joint DAMA-TCP protocol optimization through multiple cross layer interactions in DVB RCS scenario," in *Proc. IEEE IWSSC 2007*, pp. 121-125, Salzburg, Sept. 2007.
- [15] J. Alins, J. Mata-Diaz, J. L. Muñoz, E. Rendón-Morales, and O. Espqrza, "XPLIT: A cross-layer architecture for TCP services over DVB-S2/ETSI QoS BSM," *Computer Netw.*, vol. 56, no. 1, pp. 412-434, 2012.
- [16] M. K. Park and D. G. Oh, "Cross-layer design for improving TCP PEP performance in DVB-RCS2 networks," in *Proc. IEEE ICTC*, pp. 846-847, Jeju, Oct. 2013.
- [17] K. H. Lee, H. K. Jung, and J. H. Kim, "A study on PEP based on cross-layer and network coding in DVB-RCS networks," *J. IEIE*, vol. 52, no. 3, pp. 25-31, Mar. 2015.
- [18] PEPsal source code, available at: <http://www.sourceforge.net/projects/pepsal/>
- [19] DummyNet, available at: [http:// info.iet.unipi.it/~luigi/dummynet/](http://info.iet.unipi.it/~luigi/dummynet/)

김 종 무 (Jong-Mu Kim)



2015년 : 연세대학교 컴퓨터공학과 졸업
2015년~현재 : 아주대학교 전자공학과 석사과정
<관심분야> 위성 통신, WAN/위성통신 가속기, Performance enhancing proxy 등

이 규 환 (Kyu-Hyan Lee)



2007년 : 아주대학교 전자공학부 졸업
2007년~현재 : 아주대학교 전자공학과 석/박사 통합과정
<관심분야> 네트워크 코딩, 위성통신, WLAN, 무선망 QoS, Ad-hoc, Mesh network 등

Nathnael Gebregziabher W.



2008년 : Mekelle Institute of Technology Electronics and Communication Engineering 졸업
2013년~현재 : Ajou University Electrical and Computer Engineering in Master Degree

<관심분야> TCP performance in satellite communication, Performance enhancing proxy 등

김 재 현 (Jae-Hyun Kim)



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

<관심분야> QoE/QoS, 무선 MAC 프로토콜, IEEE 802.11/15, B5G 통신 시스템, 국방 전술네트워크, 위성시스템 등