WiMAX 네트워크 상에서 TCP 혼잡제어 알고리즘간 성능 비교 분석*

이문섭, 김태현
서울시립대학교 기계정보공학과
watery82@uos.ac.kr, thkim@uos.ac.kr

Performance Analysis of the TCP Congestion Control Algorithms over WiMAX Networks

Moonsup Lee, Taehyoun Kim
Dept. of Mechanical and Information Engineering, University of Seoul

요 약

최근 차세대 휴대 인터넷 기술로 주목받고 있는 WiMAX 네트워크의 성능을 향상시키기 위한 연구가 많이 진행되고 있다. WiMAX 네트워크을 이용하는 대부분의 응용들이 TCP/IP 계층을 사용하며 기존 TCP 전송기법들이 유선망을 주로 고려한 한계가 있으므로, WiMAX 네트워크 상에서의 전송 계층 성능 향상에 대한 연구의 필요성이 증대되고 있다. 본 논문에서는 실제 WiMAX 네트워크 상에서 기존 유선 네트워크에서 사용되던 TCP Reno, Veno, Westwood, Cubic 등의 TCP 혼잡제어 알고리즘들의 성능을 측정하고 비교, 분석하여 WiMAX 네트워크에서 TCP의 동작 특성을 살펴본다.

1. 서론

IEEE 802.16e 표준에 기반한 Mobile WiMAX 기술은 국내에서 WiBro라는 이름으로 상용화되었으며, 높은 전송율과 이동성을 바탕으로 차세대 이동통신 기술로써 주목받고 있다. 최근 WiMAX 네트워크의 성능 향상을 위한 많은 연구가 진행되고 있으며, 특히 MAC 계층에 초점을 맞춘 연구가 활발히 진행되고 있다.

전송 계층은 흐름 제어와 혼잡 제어를 통해 효율적이고 신뢰성 있는 전송을 보장한다. 전송 계층 프로토콜 중 TCP는 현재 인터넷에서 사실상 표준으로 사용되고 있으며, WiMAX 네트워크를 사용하는 대부분의 응용들이 기존 네트워크에서 사용되던 것임을 감안하면 TCP에서의 성능 연구는 필수적이다.

본 논문에서는 TCP성능에 중요한 영향을 미치는 혼잡제어에 초점을 맞추어, 실제 WiMAX 네트워크상에서 TCP 혼잡제어 알고리즘인 TCP Reno, Veno, Westwood, Cubic의 혼잡제어 성능을 측정하였다. 측정된 결과를 토대로 TCP의 성능이 WiMAX 네트워크에 미치는 영향을 분석하였다.

2. 관련 연구

일반적으로 네트워크 처리 용량 이상의 데이터가 전송되어 병목구간이 발생하고, 이에 따른 버퍼 오버플로우로 인해 패킷 손실이 발생하는 상황을 혼잡 상황이라고 한다. TCP의 혼잡 제어는 이러한 혼잡 상황을 탐지하고 이에 신속히 대응하여 최대의 전송 효율을 보장할 수 있도록 한다[1]. 기존의 유선 네트워크에서는 패킷 손실이 혼잡 상황에서만 발생하였으나, 무선 네트워크에서는 무선 링크의 특성으로 인한 랜덤 패킷 손실이 발생하므로 이를 혼잡 상황과 분리하여 대처하는 것이 TCP 성능에 중요한 요소이다[2][3].

TCP Reno는 유선 네트워크에서 사실상 표준으로 사용되는 혼잡제어 알고리즘으로 *cwnd* (Congestion Window)와 *ssthresh* (SlowStart Threshold) 등 두 개의 변수를 통해 혼잡 제어를 수행한다. cwnd가 ssthresh 이상이 되면 혼잡회피 단계가 되어 cwnd가 선형적으로 증가하며, 중복된 ACK 전송이 일어나거나 재전송 타임아웃이 발생하면 혼잡 상황으로 판단하여 재전송을 실행하고 cwnd 크기를 감소시켜 혼잡제어를 수행한다. TCP Reno를 기반으로 하여 무선 환경을 고려한 TCP 혼잡제어 알고리즘들이 개발되었다. TCP Veno는 매 라운드 트립 타임 (RTT)마다 큐의 백로그(Backlog)를 통해 네트워크의 다음 상태를 예측하여 혼잡 상황을 회피한다. TCP Westwood는 ACK를 이용하여 현재 가용한 대역폭을 계산하여 혼잡 상황을 회피한다. TCP Cubic은 RTT를 사용하지 않고 현재 혼잡 윈도우와 최대 혼잡윈도우를 사용하여 중간 값을 Binary Search 알고리즘을 통해 찾아 감으로써 빠르게 최대 성능을 복구할 수 있도록 한다.

3. TCP 성능 측정

실험을 위하여 본 연구진에서 개발한 WiMAX CM[4]의 SDK을 D-ITG[5] 패킷 생성기와 통합하여 WiMAX 네트워크 관리와 패킷 생성을 동시에 가능하도록 하였으며, 리눅스의 getsockopt 시스템 콜을 통해 TCP 소켓의 정보를 추적하여 TCP 계층의 정보를 수집할 수 있도록 하였다. 실험은 WiMAX 네트워크에 접속되어 있는 고정

* 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업임(No. 2009-0075978)

클라이언트가 서버로부터 데이터를 전송받는 방식으로 진행되었으며, TCP Reno, Veno, Westwood, Cubic 네 개의 혼잡제어 알고리즘에 각각 Payload 크기가 512, 4096 Bytes인 두 개의 CBR (Constant Bit Rate) 트래픽을 적용하여 처리율, cwnd, 재전송 비율을 측정한 후 비교, 분석하였다. 측정된 실험 결과는 <표 1>과 같다.

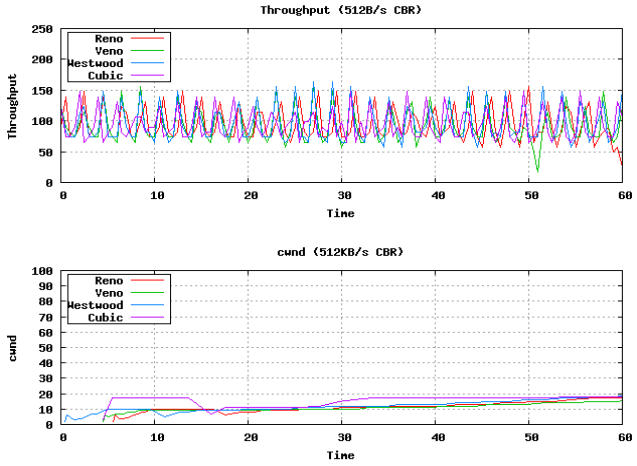
<표 1> 실험 측정 결과

	알고리즘	처리율 (KB/s)	재전송 비율
<실험 1> 512KB/s CBR	Reno	63.99	0.00049
	Veno	64.29	0.00025
	Westwood	64.32	0.00047
	Cubic	63.83	0.00023
<실험 2> 4096KB/s CBR	Reno	487.53	0.0005
	Veno	487.67	0.00025
	Westwood	469.59	0.00357
	Cubic	503.79	0.03976

4. 성능 측정 결과 분석

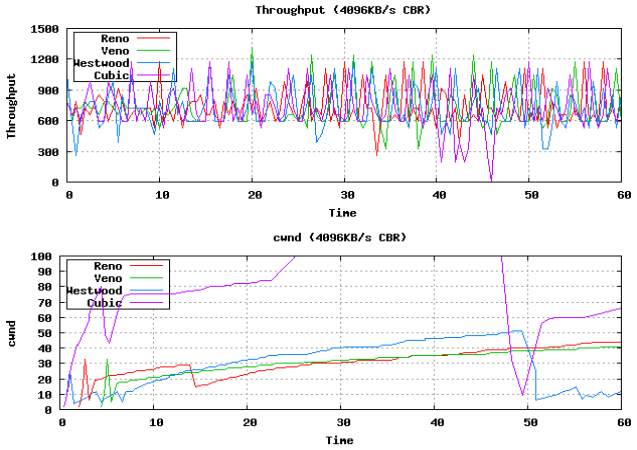
<실험 1>과 같이 트래픽의 Payload 크기가 작고 트래픽이 CBR로 일정할 때는 혼잡 상황이 거의 발생하지 않았다. 따라서 (그림 1)과 같이 혼잡제어 알고리즘 사이에 처리량에 차이가 거의 없으며 cwnd가 선형적으로 변하는 것을 볼 수 있다. 그러나 (그림 2)와 같이 Payload 크기를 4096Bytes로 하였을 때는 각 혼잡 제어 알고리즘간 cwnd 크기의 편차가 크게 나타나며 처리량의 변동폭도 증가하는 것을 볼 수 있다. <실험 1>의 경우 MSS (Maximum Segment Size)가 536 Bytes로 각 하나의 세그먼트에 전체 Payload를 포함할 수 있으나 <실험 2>에서 MSS는 1348 Bytes로 하나의 Payload를 여러 개의 세그먼트로 나누게 되므로 전체 전송 패킷 수가 증가하게 된다. 즉 Payload 크기의 증가로 인해 각 세그먼트의 크기 뿐만 아니라 세그먼트의 수도 증가하기 때문에 병목구간에서의 혼잡상황이 증가하기 때문이다.

<표 1>에서 실험 2의 경우 TCP Cubic이 많은 재전송에도 가장 좋은 처리율을 보이는 것을 볼 수 있다. (그림 2)를 보면 TCP Cubic의 경우 cwnd 크기가 빠르게 증가



(그림 1) 실험 1의 처리량과 cwnd 비교

하는 것을 알 수 있다. WiMAX 네트워크의 경우 유선 네트워크와 비교하여 긴 RTT를 가지므로[3] TCP Veno와 Westwood와 같이 RTT에 기반하여 혼잡 회피를 수행하는 것보다 TCP Cubic과 같이 RTT에 독립적인 방법이 더 유리하다.



(그림 2) 실험 2의 처리량과 cwnd 비교

5. 결론 및 향후 과제

실제 WiMAX 네트워크에서는 TCP Cubic과 같이 RTT에 독립적인 혼잡제어 알고리즘이 유리하다는 것을 알 수 있었다. 하지만 처리량이 가장 좋은 TCP Cubic과 가장 낮은 TCP Westwood 사이의 성능차이는 7.3%로 고정된 WiMAX 네트워크 환경에서는 TCP 혼잡제어 알고리즘간의 성능이 크게 차이나지 않았다.

WiMAX 표준은 MAC 계층에서 HARQ와 같이 패킷 손실을 예방하는 알고리즘을 사용하고 있으므로 이러한 알고리즘이 TCP 성능에 미치는 영향을 알아볼 필요가 있다. 또한 WiMAX 네트워크의 장점인 이동성에 대해 이동간 TCP의 성능을 파악을 위한 실험이 필요하다. 현재 WiMAX 네트워크의 이동성에 대한 TCP의 성능을 측정하고 TCP를 통한 성능향상에 관한 연구가 진행 중이다.

참고문헌

[1] J. Postel, "Transmission control protocol", RFC 793, 1981.

[2] F. Lefevre and G. Vivier, "Understanding TCP's behavior over wireless links", Communications and Vehicular Technology 2000, pp. 123-130, 2000.

[3] Y. Tian, K. Xu, and A. Nirwan, "TCP in Wireless Environments : Problems and Solutions", IEEE Radio Communications, March 2005.

[4] 이문섭, 김형석, 김익환, 김태현, "리눅스에서 WiMAX 연결 관리 및 네트워크 모니터링을 위한 소프트웨어의 설계 및 구현", 제32회 한국정보처리학회 추계학술발표대회 제출, 2009.

[5] Stefano Acallone et al., "Analysis and experimentation of Internet Traffic Generator", ICNP 2003, 2003.