

무선 랜을 사용하는 Ad-Hoc 네트워크에서 TCP와 UDP 트래픽 비교

유시환⁰ 최진희 진현욱 유혁
고려대학교 컴퓨터학과
{shyoo⁰, jhchoi, hwjin, hxy}@os.korea.ac.kr

Comparison of TCP with UDP over Multi-Hop Wireless Ad-Hoc Network Using Wireless LAN

See-Hwan Yoo⁰ Jin-Hee Choi Hyun-Wook Jin Hyuck Yoo
Dept. of CS, Korea Univ.

요 약

본 논문에서는 802.11을 기반으로 한 무선 링크에서 인터넷 상의 대표적인 전송계층 프로토콜인 TCP(Transmission Control Protocol)와 UDP(User Datagram Protocol)의 성능을 비교한다. 또한 이들 프로토콜이 무선 환경에서 보이는 문제점을 지적하고 원인을 분석한다. 무선 네트워크의 용량은 유선 네트워크와 다르며, 따라서 유선 네트워크 용량을 기반으로 설계된 TCP는 불안정하게 작동한다. 본 논문에서는 주목할만한 전송 프로토콜의 병목현상을 나타내는 원인으로 MAC(Medium Access Control)에 의해 채널을 잡기 위한 경쟁이 심각함과 전송 계층 프로토콜의 네트워크 용량 예측이 올바르지 않음을 지적하고, TCP와 UDP를 비교하여 무선 망에서도 네트워크 상황에 적응하는 기법이 중요함을 확인한다.

1. 서 론

본 논문에서는 IEEE 802.11 MAC을 사용하는 무선 Ad-Hoc네트워크에서 전송 프로토콜의 성능을 비교 분석하고 평가한다. 비교를 위해서 인터넷의 대표적인 전송 프로토콜인 TCP와 UDP를 사용하며, 무선 링크가 전송 프로토콜에 미치는 영향을 살펴보도록 한다. TCP 트래픽은 UDP 트래픽과는 달리 네트워크나 수신 노드로부터 피드백을 받는다. 무선 환경에서 TCP와 UDP의 비교를 통해 네트워크의 상황에 적응하는 것이 무선 환경에서 어떤 차이를 보이는지 확인한다. 또한 전송 계층의 신뢰성 있는 프로토콜로 인터넷 망에서 널리 쓰이고 있는 여러 가지 버전의 TCP에 대해 성능을 측정한다. 홉 수를 늘리면서 MAC과 전송 프로토콜의 동작에 대해 살펴봄으로써 Vegas를 제외한 기존의 여러 가지 TCP 버전들이 효율적으로 동작하지 않는 원인을 분석한다.

2. 연구 배경 및 관련 연구

2.1 연구 배경

IEEE 802.11표준을 사용하는 무선 랜의 경우, CSMA/CA 방식을 사용한다. 같은 셀 안의 노드들은 같은 채널을 공유하는 802.11의 경우, 한 노드가 데이터를 전송하기 위해서 미디어를 확인한 후 전송중인 노드가 없으면, 일정 시간이 지난 후 전송을 시도한다. 서로 충돌이 나지 않으며 미디어를 공평하게 공유할 수 있는 방식으로 제안된 것이 DCF이다. 802.11 WLAN의 DCF는 충돌이 발생할 때마다 충돌 원도우를 지수적으로 증가시킨다. 전송 시점은 충돌 원도우에 의해 결정되므로, 충돌이 발생하면

노드는 채널 경쟁에서 우위를 빼앗기게 된다.

두 홉의 거리를 가진 노드 간의 통신에서 송신자와 수신자는 상대방의 상태를 알지 못한다. 같은 시간에 서로 전송 시도를 하다가 충돌이 생기는 경우가 존재한다. 이를 숨은 노드 문제(Hidden Node Problem)라고 부른다. 이를 해결하기 위해서 RTS(Request To Send)/CTS(Clear To Send) 방식이 제안되었다. 그림 1은 이러한 무선 랜의 전송 방식을 도식적으로 나타낸 그림이다. 전송을 원하는 노드는 RTS 메시지를 전송하기를 원하는 노드에게 전달하고, RTS를 받은 노드는 주변의 모든 노드에게 CTS를 보내어 주위의 다른 노드들의 전송을 막도록 한다. 이를 통하여 데이터의 충돌은 많은 부분 경감된다. 하지만, 데이터 전송에 참여하지 않는 CTS를 받은 노드는 데이터를 보내지도 받지도 못하기 때문에 사용 가능한 대역폭을 낭비하게 된다. 이를 알려진 노드 문제(Exposed Node Problem)라고 부른다. 위에서 언급한 두 가지 문제는 널리 알려진 무선 랜의 한계이다.



그림 1. 무선 랜의 전송 방식

무선 랜의 연구는 현재까지 더욱 효율적인 MAC의 설계나 더욱 공평한 미디어 공유를 위한 DCF 알고리즘 등의 MAC 프로토콜 수준에서 이루어져왔다. 하지만, 실제로 성능평가에 직접적인 영향을 미치는 전송 프로토콜 수준의 성능 평가는 제대로 이루어지지 않았다. 또한 무선

Ad-Hoc 네트워크쪽의 연구는 다양한 토폴로지의 변화에 대응하기 위한 라우팅 알고리즘과 효율적인 배터리 관리 문제에 초점이 맞추어져 있었으며, 무선 링크와 관련된 전송 프로토콜의 동작의 문제를 지적인 연구는 매우 미비했다. 본 논문에서는 전송 프로토콜이 MAC의 동작에 의해 받는 영향을 살펴봄으로써 MAC의 동작과 어울리는 전송 프로토콜을 살펴본다.

2.2 관련 연구

무선 랜의 채널 경쟁으로 인한 네트워크 용량에 관한 연구는 이론적인 수준에서 많은 부분 이루어졌다[1][2]. 노드 수 n 에 따라 최적의 트래픽 상황일 때 최대 노드당 $\theta\left(\frac{W}{\sqrt{n}}\right)$ 의 처리량(throughput)을 가지는 것으로 알려져 있다. 또한 실험 결과를 통해 노드 수 n 에 따라 최적의 환경에서 최대 허용 처리량이 $\frac{c}{n^{0.68}}Mbps$ 정도로 감소함을 밝

힌 바 있다[3]. TCP나 UDP와 같은 전송 프로토콜은 MAC과는 무관한 전송 비율을 가지기 때문에 전송 프로토콜의 성능이 유선망에서 보다 크게 낮은 값을 가진다. 앞서 언급한 연구들은 전송 프로토콜과의 연관성에 대한 설명이 부족하였으며, 일부 언급한 논문에서 역시 성능 저하의 원인을 밝히지는 않고 있다[4].

무선 Ad-Hoc 환경에서 Vegas와 비교했을 때 처리량이 크게 나타났다[5]. 이는 Vegas가 혼잡 윈도우의 크기를 작은 값으로 일정하게 유지하면서 한 노드가 길게 채널을 잡지 않고 여러 노드가 무선 채널을 나눠가질 수 있도록 하기 때문이다. Reno의 경우 역시 최대 윈도우 크기를 4로 제한하면 처리량이 Vegas와 유사한 수준까지 늘어난 것을 확인할 수 있다.

3. 프로토콜 별 성능 평가 및 분석

3.1 성능 평가의 기준

성능 평가를 위해 NS-2 시뮬레이터를 사용하였다. 성능 평가의 기준으로는 프로토콜들의 일정 시간 동안 받은 패킷의 수, 네트워크에서 유실된 패킷의 수를 비교하여, 프로토콜 간 링크 활용도를 비교하도록 한다.

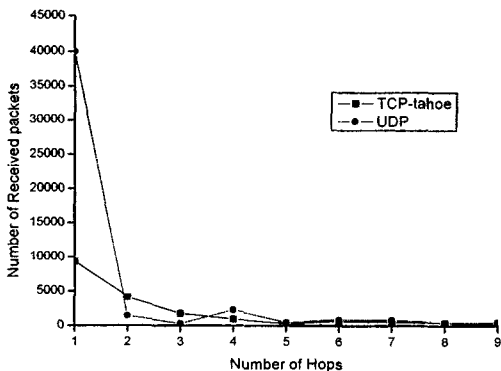


그림 2. 홑 수 증가에 따른 UDP와 TCP 전송량 변화

3.2 UDP와 TCP 전송량 비교

UDP 프로토콜은 TCP에 비해 가볍고, 사용자가 원하는 비율로 데이터를 전송한다. 초점을 맞추고자 하는 UDP와 TCP의 차이점은 TCP의 경우 ACK 패킷을 통해 역방향의 피드백을 받는 점이다. 역방향 트래픽이 존재하는 TCP와 비교하여 UDP의 전송율은 높을 것으로 예상된다. 그러나 그림 2는 시뮬레이션을 통한 UDP의 홑 수의 증가에 따른 불안정한 패킷 전송률을 보여준다.

UDP 전송 트래픽의 경우 경쟁이 없는 한 홑의 통신에서는 전혀 패킷 유실 없이 링크의 최대 사용량을 모두 사용하고 있다. 하지만, 홑 수가 늘어나면서 경쟁의 영향으로 인해 거의 모든 패킷이 유실되는 것을 관찰할 수 있다.

특히 패킷 유실은 전송을 시작한 노드와 다음 노드에 거의 집중되어 분포하는 것을 확인하였다. 이는 UDP 데이터그램이 송신 측에서 수신 측의 상황이나 네트워크의 상태를 고려하지 않고 전송을 시도하며, MAC 수준의 DCF에 전적으로 의존하기 때문이다. 더구나 어플리케이션은 사용자가 원하는 전송률로 계속 데이터를 네트워크로 보내려고 시도하므로 인터페이스 큐가 넘치는 유실이 대부분이다. 이렇게 전송 초기단계에서 유실된 패킷은 네트워크의 사용 가능한 네트워크의 대역폭을 낭비하게 되고 심지어 TCP보다도 낮은 패킷 전송률을 보인다. 2홑과 3홑의 경우 TCP의 전송량(받은 패킷 수)이 더 높으며, 홑이 커질수록 TCP와 유사한 전송량을 보인다.

3.3 TCP 버전 별 성능 평가

TCP의 경우 버전 별로 홑 수를 늘려가면서 전송량의 추이를 살펴보았다. Vegas를 제외한 현재까지 제시되었던 TCP 버전들의 혼잡제어 기법들은 무선 환경에서 큰 영향을 주지 못하는 것을 실험을 통해 확인할 수 있었다. 그림 3과 4은 홑 수의 증가에 따른 여러 가지 버전의 TCP 성능을 보여주고 있다. TCP-Vegas를 제외하면 나머지 버전들은 거의 유사한 전송량을 보이고 있다. TCP의 경우 Vegas를 제외한 버전은 전송량이나 패킷 유실이 거의 비슷한 수준에서 나타나고 있다.

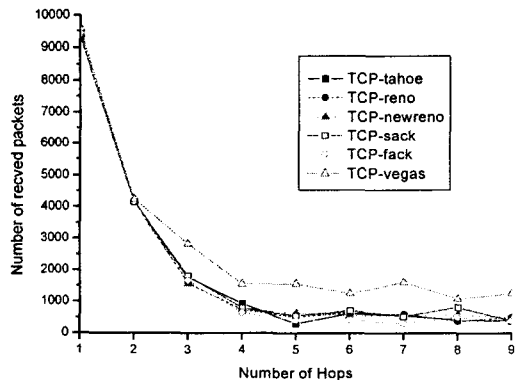


그림 3. 홑 수 증가에 따른 TCP의 버전 별 전송량 추이

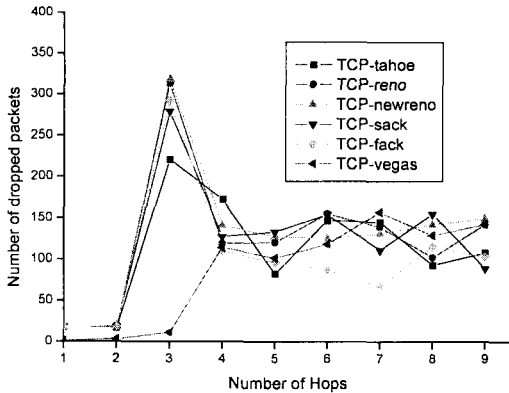


그림 4. 홑 수 증가에 따른 TCP 버전 별 패킷 유실 추이

패킷 유실의 직접적인 원인은 MAC 레벨에서 충돌로 인한 유실과 인터페이스 큐의 용량 이상의 데이터 전송으로 인한 패킷 유실이다. 데이터 패킷의 유실은 송신자와 다음 홑의 중간노드에서 많이 나타났고, ACK 패킷의 유실은 수신자와 다음 홑의 중간 노드에서 많이 나타났다. 특히 패킷의 유실은 3홑일 때 가장 높게 나타났으며, 이후로는 점차 안정적으로 동작하는 것을 볼 수 있다.

홑 수가 작을 때는 무선 네트워크의 자체보다는 노드의 프로세싱 오버헤드가 병목 현상의 원인이 된다. 점차로 홑 수가 증가함에 따라 네트워크가 병목 구간으로 옮겨진다. TCP의 송신자의 처리와 수신자의 처리를 비교하였을 때, 수신자의 프로세싱 오버헤드보다 송신자의 프로세싱 오버헤드가 크기 때문에 TCP 데이터보다 ACK의 처리 속도가 빠르다. 따라서 ACK이 하위 레이어로 빠른 속도로 전달된다. 이것이 병목구간으로 적용되며, 이 때문에 3홑의 ACK 패킷의 유실이 많아진다. 3홑일 때의 패킷 유실이 가장 많은 이유는 RTS/CTS 방식의 비추얼 캐리어 센싱 방식으로 인한 충돌 범위가 3홑일 때 가장 크게 나타나며 이후에는 일정한 수준으로 나타나기 때문이다.

Vegas는 3홑일 때에도 낮은 충돌 횟수를 보이고 있으며, 이것이 결국 전송률에 영향을 미치는 것으로 해석할 수 있다. 또한 낮은 충돌 횟수는 전송 속도를 조절함으로써 RTS/CTS의 문제점을 감소시킬 수 있음을 확인할 수 있다. 하지만 Vegas 역시 4홑 이상의 환경에서는 패킷 유실 횟수가 크게 증가하였으며, 이후에는 큰 폭으로 변동이 있음을 확인할 수 있다.

4. 무선 Ad-Hoc을 위한 전송 프로토콜의 설계 방향

TCP는 유선 환경에서 네트워크의 상황에 잘 적응하며, 신뢰성 있는 전송을 하는 특징을 가지고 있다. 무선 환경에서는 수신 측이나 네트워크로부터 피드백을 받는 것이 바람직하며, 네트워크의 용량 자체가 유선 네트워크의 지연시간-대역폭 곱으로 계산되지 않고, 채널을 잡기 위한 경쟁이 네트워크의 용량에 고려되어야 한다. 또한 참고문헌[5][6]에서 밝혔듯이 일반적인 TCP의 슬라이딩

윈도우 기법 대신 TCP의 전송률을 고려한 프로토콜이 전체 네트워크의 링크 사용을 측면에서 볼 때 유리하다. 따라서 무선 랜과 같은 링크를 사용하는 경우, 전송 프로토콜이 경쟁을 최소화할 수 있는 전송률을 찾아내고 환경에 따라 적응하는 방식이 필요하다.

5. 결론

본 논문에서는 기존의 유선 인터넷에 맞춰진 전송 프로토콜인 UDP와 TCP를 IEEE 802.11 MAC을 사용하는 무선 Ad-Hoc 네트워크 환경에서 비교하였으며 여러 가지 개선 방안에 대해 살펴보았다.

역방향 트래픽이 없는 UDP의 경우, 경쟁의 문제는 심하게 발생하지 않으나, 전송 프로토콜의 수준에서 MAC의 경쟁으로 인한 최적의 전송률을 알지 못하므로 송신 초기 단계에 많은 패킷이 유실되고 심지어 TCP보다도 낮은 데이터 처리율을 보이기도 하였다.

여러 가지 버전의 TCP로 실험한 결과 전송 프로토콜인 TCP는 무선 랜의 네트워크 용량을 이해하지 못하기 때문에 서로 비슷하게 낮은 성능을 보였다. 3홑의 환경에서 RTS/CTS 방식으로 인한 패킷 충돌로 패킷 유실이 크게 발생하였고, Vegas의 경우 3홑의 경우에도 낮은 패킷 유실을 보였다. 하지만 4홑 이상이 되면서 유실된 패킷 수가 일정하게 유지되면서 변동하는 것을 확인할 수 있었으며, 이는 위에서 언급하였듯이 전송 프로토콜이 현재 상황에 적합한 전송률을 알 수 없기 때문이다. 즉, 유선 네트워크의 용량에 기초한 전송 프로토콜의 한계는 유선 네트워크와 다른 무선 네트워크의 용량에 있다.

6. 참고 문헌

1. P. Gupta and P. R. Kumar, "The Capacity of Wireless Networks". IEEE Transactions on Information Theory, 46(2):388-404, March 2000.
2. J. Li, C. Blake, D. S. J. De Couto, H. I. Lee, and R. Morris. "Capacity of ad hoc wireless networks". In Proc. seventh Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom), pages 61-69, 2001
3. P. Gupta, R. Gray and P. R. Kumar, "An Experimental Scaling Law for Ad Hoc Networks", www.black.cs1.uiuc.edu/~prkumar
4. Venkatesh Ramarathinam and Miguel A. Labrador. "Performance Analysis of TCP over Static Ad Hoc Wireless Networks" In Proc. fourteenth IASTED International Conference on Parallel and Distributed Computing and Systems(PDCS 2002), November 4-6, 2002
5. Shugong Xu, T. Saadawi, Myung Lee, "Comparison of TCP Reno and Vegas in wireless mobile ad hoc networks". In Proc. 25th Annual IEEE Conference on Local Computer Networks (LCN'00), pages 42-43, 2000
6. Prasun Sinha, Narayanan Venkitaraman, Raghupathy Sivakumar, and Vaduvur Bharghavan. "WTCP: A reliable transport protocol for wireless wide-area networks". TIMELY Group Research Report, January 1999
7. S. Xu, and T. Saadawi, "Does the IEEE 802.11 MAC Protocol Work Well in Multihop Wireless Ad Hoc Networks?" IEEE Communications Magazine, Volume 39 Issue 6, Jun. 2001 pp. 130-137.