

휴대 인터넷에서 처리율 향상을 위한 Wireless TCP 프로토콜

A Wireless TCP Protocol for Throughput Enhancement in Wireless Broadband

* 문 일 영
* Il-Young Moon

Abstract - In this paper, we investigate the wireless TCP protocol for throughput improvement in wireless Broadband. If the burst error duration of a wireless link is significantly long, retransmissions of lost packets by Snoop TCP are fulfilled mainly not by the receipt of duplicate acknowledgement (DUPACKs) but by local timer expiration. With the proposed scheme, Snoop TCP recovers packet losses fast by shortening the interval of local retransmissions based on the channel status. From the simulation results, we can show that the proposed scheme can improve TCP throughput considerably.

Key Words : Wireless TCP, WiBro, Rayleigh fading, two-state Markov, Snoop

1. 장 서 론

최근 무선 환경과 같이 노이즈나 페이딩에 의한 전송 에러율이 높은 환경에서는 TCP 처리율이 낮아지는 문제가 발생한다. 그러나 인터넷에서 사용하는 Transmission Control Protocol (TCP) 는 전송 에러율이 낮은 유선 환경을 기반으로 만들어졌기 때문에, 무선 환경과 같이 노이즈나 페이딩에 의한 전송 에러율이 높은 환경에서는 TCP 처리율이 낮아지는 문제가 발생한다. 무선 링크에서 TCP 처리율 향상을 위한 방안으로 연결 분할 접근 방식 그리고 링크 레벨 재전송 방식 등이 제안되었다 [1]. 특히, 링크 레벨 재전송 방식 중 하나인 Snoop 프로토콜은 종단간의 무결성도 유지하면서, TCP 처리율도 향상시킬 수 있어 무선 링크에서 TCP 처리율을 향상시키는 방안으로 가장 적합하다고 알려져 있다 [2]. Snoop 프로토콜은 Base Station (BS) 에 존재하는 snoop agent가 지역 재전송을 수행하여 패킷 손실을 보완하는데, 지역 재전송은 Mobile Host (MH) 수신단으로부터 중복 ACK를 수신했거나, Snoop 지역 재전송 타이머가 만료되었을 경우에 발생한다. 또한, Snoop agent는 중복 ACK가 송신 측으로 전달되는 것을 막아, 송신 측에서 혼잡 제어를 수행하여 TCP 처리율이 저하되는 것을 방지한다. 그러나 Snoop agent의 경우 손실된 패킷이 존재하더라도 중복 ACK를 수신하지 못하면 타이머가 만료될 때까지 기다려야 하는 문제점이 있다. 본 논문에서는 중복 ACK 이외에 타이머가 만료되지 않더라도 new ACK의 번호를 확인, 적응적으로 패킷 손실을 판단하여 빠른 지역 재전송을 해주는 snoop+a 프로토콜을 제안한다. 그리고 무선 링크에서 snoop과 snoop+a 프로토콜의 성능을 컴퓨터 시뮬레이션을 통하여 비교 및 분석한다. 또한,

2.3GHz 휴대 인터넷 서비스인 Wireless Broadband 를 무선 환경으로 선택하여 TCP의 성능을 향상하고자 한다. 휴대 인터넷에서의 무선 채널 모델링에는 무선랜에서 고려하지 않았던 사용자의 이동 속도에 대한 고려가 필요하여 본 논문에서는 무선 채널을 Rayleigh 페이딩 채널로 고려하였고, 이를 two-state Markov 체인을 사용하여 모델링 하였다.

2. 장 Snoop 프로토콜

Snoop은 링크 레이어에서 동작하는 프로토콜로서 그림 3.1 은 snoop 시스템 구조를 나타내고 있다.

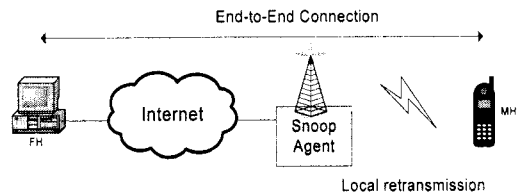


그림 1. Snoop 시스템 구조

이 방법은 BS에 TCP 패킷과 ACK 패킷의 흐름을 관찰하는 snoop agent를 두어 TCP의 종단간 특성을 유지하면서 무선 링크에서의 패킷 손실에 의한 불필요한 TCP 혼잡제어가 발생되지 않도록 하는 방법이다 [3],[4]. 이 방식에서는 BS에 위치한 snoop agent가 Fixed Host (FH) 에서 MH로 전송되는TCP 패킷을 BS 내의 버퍼에 저장한 후 수신단으로 전송한다. 또한 또 MH로부터 전송되는 ACK도 FH로 전달한다.

만약 MH로 부터 중복 ACK 패킷을 수신하게 되면 snoop agent는 중복 ACK를 FH에게 전달하지 않고 BS의 버퍼에 저장되어 있는 해당 TCP 패킷을 MH에게 지역 재전송

저자 소개

* 문일영 : 한국기술교육대학교 인터넷미디어공학부

(local retransmission) 하여 무선 링크에서의 패킷 손실을 지역적으로 복구하고 중복 ACK를 FH에게 전달하지 않음으로써 FH가 무선링크에서의 높은 Bit Error Rate (BER) 에 의한 패킷 손실을 전체 네트워크의 혼잡 상황으로 잘못 판단하여 불필요한 혼잡 제어 메커니즘을 통해 TCP 전송율을 낮추는 것을 방지한다. 또한 MH에게 전송하고 아직 ACK를 받지 못한 패킷이 snoop agent의 버퍼에 남아있고 200 ms동안 FH와 MH 양쪽에서 아무런 전송이 없을 경우 교신의 deadlock 상태로 들어간다. 이 경우에 FH에서 재전송 timeout이 발생하는 것을 미연에 방지하기 위하여 snoop 지역 재전송 타이머가 동작하여 지역 재전송을 하게 된다.

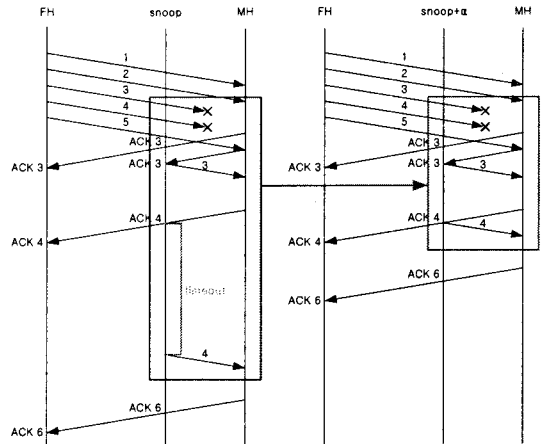
2.1 질 Snoop 프로토콜과 Snoop+a 프로토콜의 동작

그림 2는 snoop과 snoop+a 프로토콜의 동작 차이를 보여 주고 있다. 그림 2(a)는 delayed ACK가 사용된 시스템에서 snoop 프로토콜을 적용한 경우 TCP 패킷의 시간에 따른 전송 흐름도이다. 사각형 안을 보면 송신 측에서 보낸 3, 4번 패킷이 무선 링크에서 전송 중에 손실 되었다. 그러나 5번 패킷이 정상적으로 수신 측에 전송되었기 때문에, 3번 중복 ACK가 발생하였고, snoop agent는 이를 통해 3번 패킷의 손실을 인지하고 3번 패킷을 지역 재전송한다. 그리고 지역 재전송한 3번 패킷에 대응해서 발생한 4번 ACK는 new ACK 이므로 snoop agent는 버퍼에서 3번 패킷을 비우고, 4번 ACK를 송신 측으로 전달한다. 그 후 5번 이후의 ACK가 오기를 기다리게 된다. 그러나 4번 패킷도 무선 링크 상에서 손실이 되었기 때문에 더 이상의 중복 ACK는 오지 않고, 결국 snoop 지역 재전송 타이머의 만료로 4번 패킷을 지역 재전송 한다.

그림 2(b)는 제안한 snoop+a 프로토콜을 사용한 경우 TCP 패킷의 시간에 따른 전송 흐름도이다. Snoop+a 프로토콜은 중복 ACK 이외에도 new ACK로 패킷 손실을 판단할 수 있기 때문에 4번 ACK를 수신한 다음의 동작에서 snoop 프로토콜과 차이점이 있다. Snoop 프로토콜에서는 new ACK인 4번 ACK를 수신하고 단순히 FH로 전달만 해주었지만, snoop+a 프로토콜에서는 new ACK인 4번 ACK를 통해 4번 패킷의 손실을 판단할 수 있다. 그러므로 snoop+a 프로토콜은 손실된 패킷을 즉시 재전송 하여 snoop 지역 재전송 타이머의 만료를 기다리지 않고 빠른 패킷 손실 복구를 수행한다.

3. 장 휴대 인터넷 무선 환경 모델링

휴대 인터넷의 사용 주파수 대역인 2.3GHz에서 하향 링크, 상향 링크의 전송속도를 각각 3Mbps, 1Mbps로, 그리고 TCP 세그먼트의 크기를 1040 bytes로 설정했을 경우, 사용자의 이동 속도와 패킷 손실율의 변화에 따른 two-state Markov 모델의 파라미터 값은 표 1과 같다. 표 1에서 v 는 사용자의 이동속도를, P_E 는 패킷 손실율을, F 는 시스템의 페이딩 마진을 나타내며, $Burst$ 는 연결해서 손실되는 패킷의 수를 나타낸다. 표로부터 사용자의 이동 속도가 빨라질수록 $f_s T$ 값이 커지고, p_{GG} , p_{BB} 및 $Burst$ 값은 작아짐을 확인할 수 있다. 여기서, p_{GG} 는 Good 상태에서 Good 상태로, p_{BB} 는 Bad 상



((a) snoop, (b) snoop+a)

그림 2. Snoop 프로토콜과 snoop+a 프로토콜의 동작 비교

태에서 Bad 상태로의 천이 확률을 의미하며, 이 천이 확률들은 정규화된 도플러 주파수와 시스템의 페이딩 마진을 가지고 구할 수 있다. 그리고 동일한 P_E 을 가지는 무선 링크에서 페이딩이 느릴수록 ($f_s T$ 값이 작아질수록) p_{BB} 가 커짐을 확인할 수 있다. 예로써, $P_E = 0.1$ 인 무선 링크에서 $f_s T = 0.01$ 와 $f_s T = 0.64$ 에 대한 p_{BB} 는 각각 0.92685와 0.11638이고, $Burst$ 는 각각 약 13.7개와 1.1개이다. 즉, 페이딩이 느린 $f_s T = 0.01$ 때에는 Bad 상태에서 Bad 상태로의 천이 확률이 약 92.7%로 높으므로 한번 Bad 상태로 진입하게 되면 계속해서 Bad 상태에 머무를 확률이 높게 되어서 약 13.7개의 패킷이 연속해서 손실되게 된다. 반면, 페이딩이 빠른 $f_s T = 0.64$ 때에는 p_{BB} 가 약 11.6%로 낮으므로 Bad 상태로 진입하더라도 다음 패킷 전송 때에는 무선 링크 상태가 Good 상태로 천이할 확률이 훨씬 높아 연결해서 손실되는 패킷의 수가 약 1.1개가 되는 것이다. 그리고 동일한 페이딩 속도일 경우 ($f_s T$ 가 동일한 경우), P_E 가 증가할수록 패킷 손실이 연접적으로 발생함도 확인할 수 있다.

표 1 P_E 와 $f_s T$ 에 따른 Markov 파라미터

v (km/h)	$f_s T$	P_E	F (dB)	p_{GG}	p_{BB}	$Burst$
1.68	0.01	0.001	29.998	0.99933	0.32945	1.4913
		0.01	19.978	0.99752	0.75431	4.0701
		0.1	9.7732	0.99187	0.92685	13.671
13.42	0.08	0.001	29.998	0.99901	0.00824	1.0083
		0.01	19.978	0.99068	0.07729	1.0838
		0.1	9.7732	0.94035	0.46319	1.8629
107.33	0.64	0.001	29.998	0.999	0.00119	1.0024
		0.01	19.978	0.99002	0.01183	1.012
		0.1	9.7732	0.90182	0.11638	1.1317

4. 장 시뮬레이션 결과

시뮬레이션 모델은 유선 링크에 있는 Fixed Host (FH) 와

무선 링크에 있는 Mobile Host (MH) 가 각각 1개인 경우에 대해서 시뮬레이션을 수행하였다. 각 노드가 1개인 경우 데이터 링크 계층에서의 접근을 위한 경쟁이 발생하지 않으므로 매체에서의 패킷 충돌에 의한 패킷 손실은 발생하지 않는다. 그러므로 무선 링크에서의 패킷 손실만이 TCP의 처리율에 영향을 주게 된다. TCP 세그먼트의 전송은 FH에서 MH로의 전송만을 가정하였고 기지국에서는 유한한 DropTail 버퍼를 사용하였다. 유선 링크는 100ms의 전송지연을 가지며, 무선링크는 64 μ s의 전송지연을 가진다고 가정하였다. TCP/IP 하위의 링크 레이어 계층은 유선 링크에서는 10Mbps 대역폭을 가지는 Ethernet을, 무선 링크에서는 하향 3Mbps, 상향 1Mbps의 대역폭을 가지는 2.3GHz 휴대 인터넷 무선 환경으로 설정하였다. 그리고 무선 링크에서의 패킷 손실은 하향 링크에서만 발생한다고 가정하였다. 전송 계층 프로토콜은 TCP-Tahoe를 사용하였고, 최대 세그먼트 크기는 1040 bytes로 설정하였다. 시뮬레이션은 무선 링크에서의 패킷 오를 P_E 를 변화시키면서 150초 동안 수행하였고, random seed를 사용하여 10회 반복한 값의 평균을 취하였다. 그리고 본 논문에서는 TCP 처리율을 비교하는 척도로 MH에 올바르게 전달된 패킷의 수를 사용하였다.

그림 3은 Base Station (BS) 에 snoop/snoop+a agent를 사용했을 때와 안 했을 경우, 각각의 P_E 에 따른 TCP 처리율을 비교한 그래프이다. BS에 snoop 및 snoop+a agent를 적용한 경우 BS에서의 지역 재전송의 결과로 FH에서 slow start가 발생하는 것을 억제할 수 있었고, TCP 처리율이 향상되는 것을 확인할 수 있었다. 그리고 두 프로토콜 모두 P_E 가 높은 무선 링크에서 더 높은 TCP 처리율 향상을 보였다.

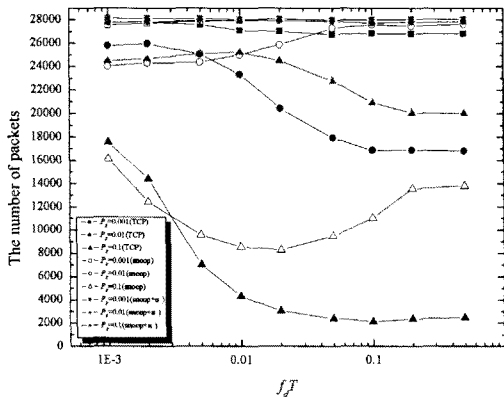


그림 3 P_E 의 변화에 따른 TCP 처리율 비교

그러나 snoop 프로토콜의 경우에는 두 가지 문제점이 발생하였다. 첫 번째 문제점은 느린 속도의 페이딩에서는 순수하게 TCP 프로토콜만을 사용한 경우보다 낮은 TCP 처리율을 나타냈다. 그리고 두 번째 문제점은 P_E 가 0.1 정도로 높은 경우에 중간 속도의 페이딩 환경에서의 TCP 처리율이 느린 속도의 페이딩이나 빠른 속도의 페이딩 환경에서보다 낮아지는 문제점이 발생하였다. 하지만, 제한한 snoop+a 프로토콜을 적용한 경우에는 중복 ACK 이외에도 new ACK를 통

해 적용적으로 지역 재전송이 가능하기 때문에 느린 페이딩에서도 TCP만을 사용한 경우보다 높은 TCP 처리율을 나타내었고, P_E 가 높은 경우에도 중간 속도의 페이딩에서 TCP 처리율이 저하되지 않아서 snoop 프로토콜의 두 문제점을 해결할 수 있었다.

5. 장 결론

본 논문에서는 2.3GHz 휴대 인터넷 무선 환경에서 사용자의 이동성을 나타내는 PE 를 변화시키면서 BS에서 snoop 및 snoop+a 프로토콜을 사용했을 때의 TCP 처리율을 시뮬레이션을 통해 비교·분석하였다. Snoop 프로토콜은 중복 ACK가 송신 측으로 전달되는 것을 막아, 송신 측에서 혼잡 제어를 수행하여 TCP 처리율이 저하되는 것을 방지하는 효과적인 프로토콜이다. 그러나 snoop 프로토콜의 경우 snoop 버퍼 내에 손실된 패킷이 존재하더라도 중복 ACK를 수신하지 못하면 타이머가 만료될 때까지 기다려야 하는 문제점이 있다. 본 논문에서는 중복 ACK 이외에도 new ACK를 사용해서 중복 ACK를 수신하지 않은 경우에도 타이머 만료를 기다리지 않고 적용적으로 패킷 손실의 여부를 판단하여 패킷을 지역 재전송할 수 있는 snoop+a 프로토콜을 제안하였다. 시뮬레이션 결과로부터 snoop+a 프로토콜이 기존의 snoop 프로토콜 보다 빠르게 패킷의 손실을 복구할 수 있음을 확인할 수 있었다. 그러므로 휴대 인터넷과 같이 고속 데이터 전송을 제공하면서 사용자의 이동성까지 지원하는 시스템에서 snoop+a 프로토콜을 사용하면 사용자의 이동 속도와 상관없이 일정 수준의 Quality of Service (QoS) 를 보장할 수 있을 것이다.

참고 문헌

- [1] H. Balakrishnan, S. Seshan, and R. H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," ACM Wireless Networks, vol. 1, no. 4, pp. 469-481, Nov. 1995.
- [2] H. Balakrishnan, S. Seshan, and R. H. Katz, "A comparison of mechanisms for improving TCP performance over wireless links," IEEE/ACM Transactions on Networking, vol. 5, no. 6, pp. 756-769, Dec. 1997.
- [3] J. Hu, G. Feng, and K. L. Yeung, "Hierarchical cache design for enhancing TCP over heterogeneous networks with wired and wireless links," IEEE Transactions on Wireless Communications, vol. 2, no. 2, pp. 205-217, Mar. 2003.
- [4] K. Ratnam and I. Matta, "WTCP: an efficient mechanism for improving TCP performance over wireless links," Third IEEE Symposium on Computers and Communications (ISCC 98), pp. 74-78, 1998.