

# Proxy TCP ACK를 이용한 무선 환경에서의 TCP 성능향상 연구

강윤식, 장주욱  
서강대학교 전자공학과

## Improvement of TCP Throughput using Proxy TCP ACK over Wireless Connection

Yoonsik Kang, Ju-wook Jang  
Dept. of Electronic Engineering, Sogang Univ.  
armani@eeca1.sogang.ac.kr  
jjang@ccs.sogang.ac.kr

### 요약

TCP 프로토콜은 데이터를 신뢰성 있게 전송할 수 있다는 장점을 가지고 있는 반면, Congestion Avoidance Scheme이 무선 환경에서는 성능저하 요인(Congestion Window Size의 감소, Timeout Period의 기하급수적 증가)으로 작용하여 TCP Throughput저하를 가져오게 된다. 이런 문제점을 해결하기 위한 기존 연구들은 TCP Semantics를 위반하거나 Router에 과도한 부하를 가하는 단점이 있다. 본 논문에서는 Snoop방식과 M-TCP방식의 장점을 결합한 새로운 기법을 제안하고 그 효율성을 실험을 통해 입증하였다. 그 결과 비트 에러율에 따라 TCP Reno보다 25% - 93%의 전송 효율 개선을 이루었다.

### 1. 서론

무선환경에서의 TCP 연결이 가지는 가장 큰 특징은 1) 전송로의 비트 에러로 인한 재전송이 일어나거나 2) Long Disconnection에 의해 Burst Packet Error를 유발할 수 있다는 것이다. 이는 유선 환경에서의 정체에 의한 패킷 손실과는 다르다. TCP는 무선 환경에서의 비트 오류에 의한 패킷손실의 원인을 Congestion으로 간주하여 Congestion Window를 감소시켜 그 결과 TCP Throughput이 크게 떨어질 수 있다[5].

무선 환경에서 TCP의 성능저하를 극복하고자 개발된 기법으로는 Snoop[2], I-TCP[2], MTCP[4]과 M-TCP[6] 등이 있으며 표 1은 이들의 장단점들 나타내었다.

	빠른 재전송	TCP Semantics	Buffer problem	Fading	Fault Tolerance
I-TCP& MTCP	O	X	X	X	X
Snoop	O	O	X	X	X
M-TCP	O	O	O	O	X

표 1 기존의 무선 프로토콜의 특징

본 논문에서는 Snoop 프로토콜의 빠른 재전송과 M-TCP의 전송률 저하 방지를 유지하면서도 Fading에 강하고 Fault Tolerant한 새로운 기법을 제안한다. 제안된 기법인 Proxy TCP<sup>1)</sup>의 Fast Retransmission과 TCP Semantics 유지, 전송 효율, 선택적 사용 가능 여부 등을 실험을 통하여 기존 방식들과 비교 분석하였다. 그 결과 64kbps<sup>-1</sup>의 에러 상황에서 Proxy TCP가 TCP Reno보다 93%의 성능 개선을 128kbps<sup>-1</sup>에서 25%의 개선을 보였다.

### 2. 무선 환경에서의 TCP의 문제점

재전송은 Congestion Window 감소와 Timeout 증가를 유발시켜 전송률을 줄이게 된다. 유선환경에서의 재전송은 네트워크 정체에 의한 패킷 손실에 의해 발생하며 무선환경에서는 무선 연결 부분에서의 비트 에러와 Disconnection, Fading에 의해 주로 패킷이 손실되어 경우에 재전송이 발생한다.

무선 환경의 비트 오류와 Disconnection에 의한 패킷 손실은 유선 환경에서 발생하는 대역폭 감소에 의한 손실

1) Fading(Long Disconnection) 발생시 대신 TCP ACK를 보내는 방식으로 Proxy TCP라 이름한다.

과 다르며 무선 환경에서 TCP 전송자의 Throughput 저하[5]는 비트 오류에 의한 손실을 정체에 의한 손실로 전송자가 오인하기 때문에 발생한다. 그 결과 TCP 전송자는 재전송 타임아웃을 2배 증가시키고 Congestion Window의 감소시켜 전송로의 효율을 떨어뜨리며 심한 경우에는 Serial Timeout을 일으키고 접속이 재개되어도 전송자는 Timeout Period를 기다려 심각한 대역폭 낭비를 가져오게 된다[6].

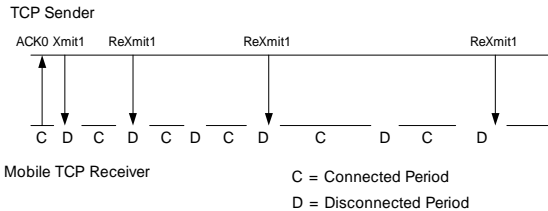


그림 1 Serial Timeout

### 3. Mobile TCP 기법과 Proxy TCP

#### 3.1 기존 Mobile TCP 기법

TCP를 무선 상황에 적합하도록 개선한 기법들은 무선 네트워크와 유선 네트워크를 분리하여 제어하는 방식(Split-Connection)과 재전송을 중간에서 대신해주거나 데이터의 전달여부를 미리 알아내어 전송자에게 전달하는 대리자를 두는 방식(Snoop, M-TCP)으로 구분할 수 있다.

##### 1) Split Connection

Split Connection 방식은 무선과 유선, 서로 다른 성질의 네트워크를 분리하여 제어함으로써 제어의 효율성을 증대시키기 위한 방안이다[7]. 대표적인 기법으로는 I-TCP[1], MTCP[4] 등이 있다.

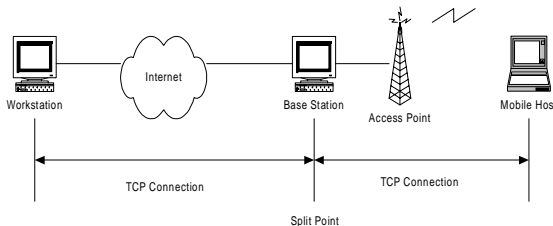


그림 2 Split Connection Protocol

I-TCP와 MTCP는 유선 호스트와 매개자간을 TCP로 연결을 하고 또 매개자와 무선 호스트 사이에 별도의 TCP를 연결하며 재전송을 매개자가 담당한다. 단점은 TCP Semantics를 유지하지 않고, 응용프로그램의 적용이 어려우며 많은 양의 버퍼가 요구된다는 점, 라우터의 작업량이 너무 많다는 점을 들 수 있고, 데이터의 저장기 수신단의 상황과는 무관하게 이루어지게 되기 때문에 버퍼가 고갈되어 패킷이 손실되는 상황이 발생할 수 있다[3].

##### 2) Snoop Protocol

유선 환경과 무선 환경이 혼재되어 있는 네트워크에서

가장 주목받는 프로토콜의 하나이다. Snoop는 라우터에 각 Flow에 관한 버퍼를 할당하고 버퍼에 전송 패킷을 저장하여 Duplicated-ACK가 도달하는 Flow에 대해서는 라우터가 대신 재전송을 해주므로써 Fast-Recovery를 구현하고 있다[2]. Snoop의 장점으로는 TCP Semantics를 유지하고 Fast-Recovery가 가능하며 전송로 효율 증대를 들 수 있고 단점은 Split Connection 방식과 같이 구현이 어렵고 라우터에 많은 작업과 버퍼가 필요하다는 점, Fading 현상에 대응방안이 없다는 것이다.

##### 3) M-TCP

M-TCP는 Cellular Network을 기반으로 하여 개발된 프로토콜로 Supervisor Host가 Advertised Window를 사용하여 전송자의 Congestion Window 감소를 억제하는 특징을 갖고 있다[6]. 장점으로는 Split Connection 방식이면서도 End-to-End TCP Semantics를 따르고 있다는 것과 Fast Recovery가 가능하다는 것이며 단점으로는 무선 호스트는 별도의 프로토콜과 Connection을 유지해야 한다는 점, 특화된 Supervisor Host를 필요로 한다는 점, Supervisor Host가 오동작을 일으키면 Connection 자체가 불가능하다는 점등을 들 수 있다.

##### 3.2 제안된 기법(Proxy TCP)

제안된 TCP 전송 기법(Proxy TCP)은 무선 네트워크와 유선 네트워크가 연결된 상황에서 Snoop의 빠른 재전송과 M-TCP의 전송률 유지 기법을 결합하여 TCP 성능 저하의 개선을 목적으로 하고 있다. Proxy TCP의 설명에 앞서 용어를 정의하고자 한다.

- Proxy Manager: Proxy TCP가 동작하는 호스트
- Proxy ACK: Proxy Manager가 전송자에게 보내는 Advertisement Window = 0 으로 설정된 ACK
- Test Packet: Proxy Manager가 무선 호스트에게 전송하는 Packet, ACK가 전달되지 않은 데이터 패킷 중 가장 오래된 것으로 한다.
- Hold State: Proxy ACK가 전송된 상태

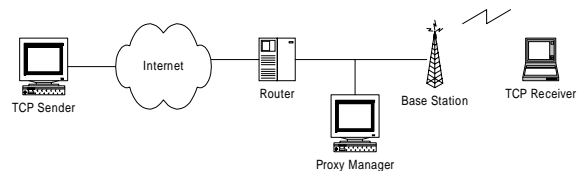


그림 3 제안된 실험 시스템

Proxy TCP는 Base station과 동일한 LAN상에 위치하는 호스트에 이식이 되며 이 호스트(Proxy Manager)는 무선과 유선간의 TCP 패킷 흐름을 관찰하여 무선 전송로의 비트 오류 발생시 재전송을 대신하고 Fading 발생시 전송자의 Congestion Window를 고정시키는 역할을 담당

하게 된다. Proxy TCP는 TCP 양단에서 전달되는 패킷이 중도에서 제거하는 상황을 유발하지 않는다. 그림 4는 Proxy Manager가 동작하는 원리를 흐름도로서 표현한 것이다.

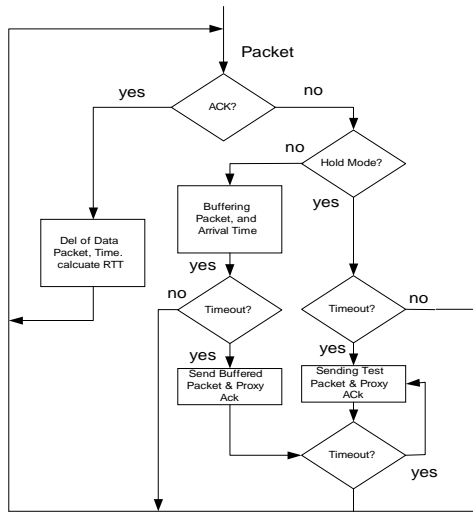


그림 4 Proxy Manager의 기능

Proxy TCP의 특징은 다음과 같다.

- 1) 간접적으로 TCP Connection에 개입한다.
- 2) 에러, Fading에 의한 패킷 손실시 전송자의 Congestion Window 감소를 억제할 수 있다.
- 3) 패킷을 중간에서 제거하지 않는다.
- 4) TCP Semantics를 유지한다.
- 5) Proxy Manager는 Connection과 무관하다.

본 논문에서 제안하고 있는 Proxy TCP의 장점은 1) 비트 에러 발생시 Fast Recovery 가능, 2) Fading 발생시 Congestion Window Size 유지, Timeout 증가 억제 3) TCP Semantics 유지, 4) 응용프로그램 사용 용이, 5) Proxy Manager의 개선 용이와 선택적 사용이 가능하다는 점, 6) 오동작으로 인한 패킷손실 방지 등을 들 수 있다. Snoop가 갖는 빠른 재전송에 M-TCP가 갖는 Fading 극복을 주목함으로써 무선환경에 적합한 기능을 구현하고 있다. 표 2는 Proxy TCP와 기존의 Mobile TCP를 비교한 것이다.

	Snoop	I-TCP & MTCP	Proxy TCP (제안된 기법)
End-to-End Semantics	O	X	O
High Bit Error	O	O	O
buffering problem	O	X	O
Fading	X	X	O

표 2 Proxy TCP와 다른 Mobile TCP와의 비교

#### 4. 구현 및 결과

그림 5의 Topology를 사용하여 무선 환경에서의 TCP Reno와 Proxy TCP 성능을 알아보고 무선 호스트에 임의로 Fading현상을 유발시켜 Proxy TCP와 TCP Reno간의 성능 차이를 비교하였다.

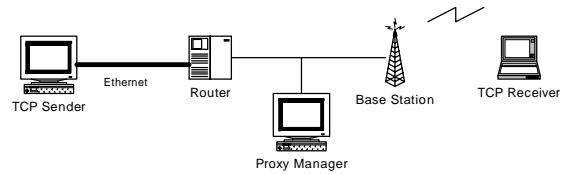


그림 5 실험 Topology

실험 환경의 유선 전송로는 10Mbps이고 무선 전송로는 2Mbps이며 각 단에서의 시간지연은 20msec이며 TCP Sender는 TCP Reno이며 비트 에러율은  $64\text{kbps}^{-1}$ 이다.

그림 6은 무선환경에서의 TCP Reno와 Proxy TCP간의 비교이다.

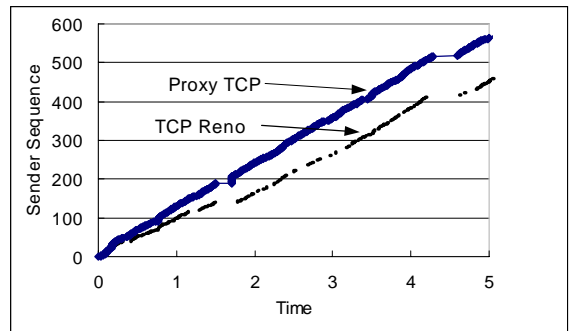


그림 6 TCP Reno와 Proxy TCP 성능 비교

그림 6에서 Proxy TCP가 무선 환경에서 TCP Reno보다 더 나은 성능을 보이고 있다. 5초동안 진행된 시점에서 전송자의 Sequence Number는 약 90의 차이가 있다. 1.6-1.7초와 4.3-4.5초 사이의 구간에서 Fading현상이 발생하였다. Fading 현상이 발생하는 구간에서 TCP Reno와 Proxy TCP의 증가속도 차이가 분명해진다.

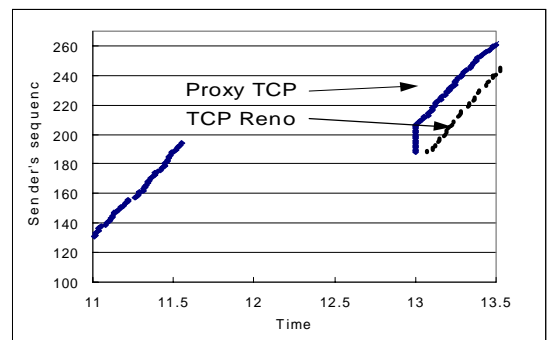


그림 7 1.5초간의 Fading 영향 비교

그림 7과 8은 Fading에 대한 반응을 자세히 알아보기

위한 실험이다. 그림 7은 Fading이 1.5초간 진행되었을 때의 영향이다. Proxy TCP의 경우 Sequence Number의 증가가 빠르게 이루어지는 것을 알 수 있다. Fading이후 TCP Reno는 Slow Start를 재개하는 반면 Proxy TCP는 Fading이전의 전송률을 갖고 전송을 재개하는 것을 확인할 수 있다.

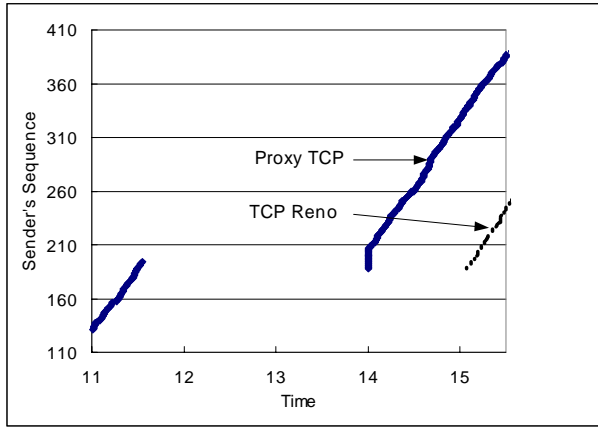


그림 8 2.5초간의 Fading 영향 비교

그림 8은 Fading이 2.5초간 유지되었을 때의 비교이다. TCP Reno는 Connection이 재개된 사실을 1초간 인식 못하고 있는 것을 확인할 수 있다. 그림 7과 비교하면 Fading이 길어질수록 Sequence Number의 증가 속도 차이가 많아지는 것을 알 수 있다.

표 2는 에러 발생 상황에 따른 TCP Reno와 Proxy TCP간의 Throughput을 비교한 것이다.

프로토콜	비트 에러율	
	128kbps <sup>-1</sup>	64kbps <sup>-1</sup>
Throughput		
TCP Reno	1.18(Mbps)	0.67(Mbps)
Proxy TCP	1.43(Mbps)	1.39(Mbps)

표 3 Throughput 비교

에러가 많은 환경에서 TCP Reno와 Proxy TCP의 성능 차이가 뚜렷해지는 것을 알 수 있다. 64kbps<sup>-1</sup>의 비트 에러 상황에서는 대략 2배의 성능 차이를 보인다.

## 5. 결론

무선환경은 TCP의 현저한 성능저하를 유발시키며 이는 High Bit Error rate와 Fading 현상 때문에 발생한다. 이를 극복하기 위한 노력으로 I-TCP, MTCP, Snoop Protocol등이 있으나 라우터에 집중되는 과도한 부하와 많은 버퍼의 사용, Fading이 길어질 때 발생하는 Serial Timeout, TCP Semantics의 준수 등 많은 문제점을 내포

하고 있다. 본 연구에서는 이러한 문제점들을 극복하기 위해 Proxy TCP를 제안하였다. Proxy TCP는 높은 에러환경과 Fading 현상에 적절히 대응하여 Fast Recovery가 가능하며 TCP Semantics를 유지하고 응용 프로그램의 사용이 용이하다. 또한 Proxy Manager와는 무관하게 TCP Connection이 이루어지므로 선택적 사용이 가능하고 Proxy Manager가 오동작을 일으키는 상황에서도 TCP Connection에는 영향을 주지 않는 장점을 지니고 있다.

## 6. 참고문헌

- [1] A. Barke and B. R. Badrinath, "I-TCP: Indirect TCP for Mobile Hosts", Proceedings of the 15th International Conference on Distributed Computing System, 1995, pp136-143
- [2] H. Balakrishnan, S. Seshan, and Randy Katz "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks", Wireless Networks, Vol. 1, No. 4, 1995,
- [3] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and Randy Katz, "A Comparison of Mechanisms for Improving TCP Performance over Wireless Links", ACM SIGCOMM'96, pp256-269
- [4] R. Yavatkar and N. B. Hagawat, "Improving End-to-End Performance of TCP over Mobile Internetworks", IEEE 1994 Workshop on Mobile Computing System and Application
- [5] A. Barke and B. R. Badrinath, "Indirect Transport Layer Protocols for Mobile wireless Environment", Mobile Computing, eds. T. Imielinski and H. F. Korth, Kluwer Academic Publishers, 1996, pp229-252
- [6] Kevin Brown and Suresh Singh, "M-TCP: TCP for Mobile Cellular Networks", ACM CCR Vol. 27, No. 5, 1997

저자 연락 정보

논문제목	Proxy TCP ACK를 이용한 무선 환경에서의 TCP 성능향상 연구
저자명	강윤식, 장주욱
논문 분야	(20) 인터넷, (3) 데이터 통신망, (18) 통신 운용관리
주소	(우 121-742) 서울특별시 마포구 신수동 서강대학교 전자공학과 대학원 컴퓨터 구조연구실 강윤식
	(우 121-742) 서울특별시 마포구 신수동 서강대학교 전자공학과 장주욱 교수님
전화/팩스	전화: 강윤식(02-3272-3220), 장주욱(02-705-8472) 팩스: 02-3272-3220
전자우편	강윤식: armani@eeca1.sogang.ac.kr 장주욱: jjang@ccs.sogang.ac.kr