

---

# 무선 TCP 성능 향상을 위한 SACK-SNOOP 프로토콜

안치현\* · 김형철\* · 우중정\*\* · 김장형\*\*\* · 이대영\*\*\*\* · 전계석\*\*\*\*

## SACK-SNOOP Protocol for Wireless TCP Performance Improvement

Chi-hyun Ahn\* · Hyung-Chul Kim\* · Jongjung Woo\*\* · Jang Hyung Kim\*\*\* · Dae-young Lee\*\*\*\* · Kye-suk Jun\*\*\*\*

### 요 약

TCP는 무선링크에서 무선망의 특징으로 인한 패킷 손실을 혼잡으로 인식하여 성능저하를 일으킨다. 이를 개선하기 위해 제안된 다양한 무선 TCP 방법중에서 SNOOP은 무선구간에서 지역적 재전송을 통해 FH에서의 빠른 재전송이나 혼잡제어를 방지함으로써 TCP의 성능을 향상시킨다. 하지만 SNOOP은 무선구간에서 연집오류를 처리하는데 있어서 개선해야 할 부분이 있다. 본 논문에서는 Freeze-TCP의 ZWA 메시지를 이용해서 FH에서 타임아웃이나 혼잡제어 발생을 막고 무선구간에서 손실된 패킷을 재전송하는 시간을 보호하면서 지역적 연집오류 재전송을 위한 방법으로 SACK-SNOOP을 제안한다. SACK-SNOOP은 기존의 SACK을 개선하여 오류환경에 따라 ACK에 포함될 오류 패킷의 시퀀스 번호의 개수를 줄임으로써 ACK의 생성과 전송, 해석에 따른 처리시간을 향상시켜 무선구간의 지역적 연집오류의 재전송을 위한 충분한 시간을 확보할 수 있다. 또한 재전송시의 오류에 대비해 FH의 지연시간을 연장함으로써 능동적으로 재전송 오류에 대처한다. 제안 방법은 실험을 통해 연집오류에 의한 패킷손실에서 기존의 방법보다 효율성이 향상됨을 검증하였다.

### ABSTRACT

Wireless network has high BER characteristic because of path loss, fading, noise and interference. Many packet losses occur without any congestion in wireless network. Therefore, many wireless TCP algorithms have been proposed. SNOOP, one of wireless TCP algorithms, hides packet losses for Fixed Host and retransmits lost packets in wireless network. However, SNOOP has a weakness for burst errors in wireless network. This paper proposes the SACK-SNOOP to improve TCP performance based on SNOOP and Freeze-TCP that use ZWA messages in wireless network. This message makes FH stop sending packets to MH. BS could retransmit error packets to MH for this time. SACK-SNOOP use improved Selective ACK, thereby reducing the number of packet sequences according to error environment. This method reduces the processing time for generation, transmission, analysis of ACK. This time gain is enough to retransmit local burst errors in wireless link. Furthermore, SACK-SNOOP can manage the retransmitted errors by extending delay time to FH. The simulation shows that our proposed protocol is more effective for packet losses in wireless networks.

### 키워드

무선 TCP, SNOOP, Freeze TCP, Mobile IP

---

\* 열린사이버대 학교

\*\* 성신여자대학교

\*\*\* 제주대학교

\*\*\*\* 경희대학교

## I. 서론

우리가 사용하고 있는 인터넷은 TCP/IP 프로토콜이 중심이며, 전송 오류가 발생하지 않는 유선망 환경에서는 효율적으로 동작하도록 설계되었다. 하지만 오늘날 인터넷은 이동통신 사업자의 서비스 확대와 가입자 수의 증가, 인터넷 기술의 급속한 발달로 음성 위주의 무선 통신에서 데이터 중심으로 전환되고 있으며 유선과 무선이 혼합된 망의 형태로 변화하고 있다.

무선망은 경로손실(path loss), 페이딩(fading), 잡음(noise), 간섭(interference) 등의 이유로  $10^{-2}$  ~  $10^{-4}$ 의 높은 BER(Bit Error Rate)을 가지며 혼잡상황과 무관하게 많은 패킷 손실이 발생하고 있으며[1][2][3], 유선망과 달리 연집오류도 많이 발생한다. 그런데 현재 유선 TCP는 알고리즘은 무선망에서 문제점이 발생하여도 유선상의 혼잡으로 인식하여 느린 시작(Slow start)[4], 빠른 재전송(Fast retransmission) 등의 불필요한 혼잡제어를 하게 된다[5][6]. 이와 같은 TCP의 혼잡제어는 링크 대역폭 이용률의 감소를 초래하여 성능저하를 일으키게 된다. 그래서 전송오류가 상대적으로 큰 무선망에서 TCP 프로토콜의 성능 개선을 위해 I-TCP[7], M-TCP[8], Freeze-TCP[9], SNOOP[10] 등 다양한 무선 TCP들이 연구 및 제안되고 있다. 본 논문에서는 이러한 무선 TCP의 제안들을 살펴보고 연집오류를 효과적으로 극복하는 새로운 방식의 SACK-SNOOP을 제안한다.

SACK-SNOOP은 SACK를 개선하여 오류 환경에 따라 전송되는 패킷의 시퀀스 번호를 줄여 ACK의 생성과 전송, 그리고 해석을 빠르게 처리되도록 하였고 Freeze-TCP의 ZWA(Zero Window Advertisement) 메시지를 이용해서 새로운 FT 시간을 정의하여 FH에서 타임아웃이나 혼잡제어 모드로 동작하는 것을 지연한다. 또한 재전송 오류에 대비해서 FT 시간을 재연장함으로써 무선구간에서 연집오류 패킷들의 재전송 시간을 보호하여 연집오류를 효율적으로 처리한다.

## II. 관련연구

### 2.1 무선 TCP

무선망을 포함하는 네트워크 구성을 보면 그림 1과 같이 FH(Fixed Host), BS(Base Station), MH(Mobile Host)

로 구성되어 있다. BS와 FH간의 유선백본(wired backbone)상에는 오류가 거의 없는 대역폭이 큰 유선링크이고 BS와 MH 간에는 링크 오류가 많고 대역폭이 작은 무선링크이다. 무선망은 경로손실, 페이딩, 잡음, 간섭 등이 발생하기 때문에 높은 BER의 특성을 가지게 된다. 또, 무선링크의 핸드오버 발생 시에도 FH와 MH간의 전송로를 연결해주는 BS가 바뀌게 되면서 순간적으로 전송이 자주 단절된다. 만일 전송 대역폭 이상의 패킷 전달 요구에 의하여 혼잡하게 되면 라우터는 감당하지 못한 패킷을 버리게 된다. 패킷을 전달하고자 했던 FH는 타임아웃에 걸리거나 패킷을 받아야 하는 MH로부터 재전송 요구를 받게 되므로 FH는 혼잡제어모드로 동작한다. 혼잡제어가 발생하면 절반씩 줄어드는 윈도우 크기에 의하여 전송 속도가 떨어진다. 무선망의 경우 무선구간의 패킷오류가 발생하면 FH가 혼잡제어 모드로 동작하는 문제점이 발생한다. 유선 TCP는 무선망의 빈번한 패킷오류와 혼잡에 의한 패킷오류를 구분 할 방법이 없다. 따라서 전송 대역폭 자체가 줄어든 것이 아니라 무선망의 특성에 의해 발생한 패킷손실을 혼잡에 의한 패킷오류로 인식하여 혼잡제어를 하게 됨으로써 FH는 전송 속도를 낮추게 된다. 그래서 오류가 잦은 무선망에서 TCP 성능을 크게 저하시킨다.

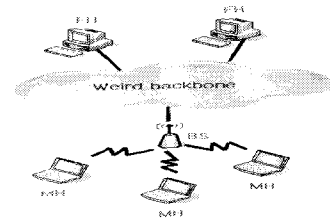


그림 1. 무선망을 포함하는 토폴로지  
Fig 1. Topology including Wireless Network

유선 TCP의 이러한 문제점을 해결하기 위해 여러 가지 무선 TCP가 제안되었다. 무선 TCP는 크게 3가지의 접근 형태를 가진다.

○ Pure-link level approach : 이러한 접근 방법은 일반적인 end-to-end TCP 연결을 기본으로 하고 있으며 FEC, ARQ 재전송 등에 사용된다.

○ Split connection approach : 이러한 접근 방법은 TCP 연결을 BS를 기준으로 유선 구간과 무선 구간의 연결로 구분하여 서로 독립적인 흐름 제어를 수행한다. 따라서 기존

의 end-to-end TCP 프로토콜의 의미를 훼손한다는 단점이 있다. 대표적인 방법에는 I-TCP[7], M-TCP[8] 등이 있다.

○ TCP-aware link layer : 이러한 접근 방식은 기본적으로 BS에 버퍼를 두어 중복 ACK를 받으면 FH측에 중복 ACK가 도착하기 전 BS가 중간에서 이를 차단하고 링크 레벨에서 처리하는 방식이다. 따라서 BS가 중복 ACK의 헤더를 열어보고 패킷을 재전송해주어야 하기 때문에 헤더 분석 능력을 가져야 한다. 대표적인 방식으로 SNOOP[10]이 있다.

### 2.2 Freeze-TCP

Freeze-TCP는 핸드오버나 페이딩 신호 세기의 약화 또는 무선 환경의 특수한 상황에 의하여 통신 두절이 임박했을 때 패킷 손실을 방지하기 위한 사전작업으로 곧 있을 연결중단에 대한 컨트롤을 MH에 넘기는 것이다 [9]. Freeze-TCP는 이를 위한 사전작업을 TCP 수신측에서 수행함으로 송신측의 변경이나 I-TCP나 SNOOP 방식처럼 중간노드인 BS가 필요하지 않다.

Freeze-TCP는 수신측 MH와 송신측 FH의 연결 해제에 예상되면, 적어도 1개 이상의 ACK에 윈도우 크기를 0으로 설정하여 ZWA를 보낸다. 이를 통해 FH는 MH가 패킷을 받을 수 없는 상황임을 알고 패킷 전송을 중지하고 해당 세션과 관련된 모든 타이머의 동작을 일시 중지시킨다.

연결 중단 전 이러한 ACK를 보내는 것이 관건인데 경고 주기(warning period)은 최소 하나의 ZWA 세그먼트가 송신 측에 도착할 수 있도록 충분히 길어야 한다. 실험에 의해 합리적인 시간은 RTT(Round Trip Time)이다. 또한 FH는 이 상태에서 MH가 윈도우를 다시 열 때까지 ZWP(Zero Window Probes)를 주기적으로 수행한다.

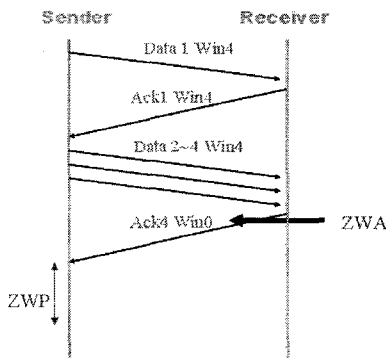


그림 2. Freeze TCP의 ZWA 전송  
Fig 2. ZWA message transmission of Freeze TCP

지수분포로 증가하는 ZWP의 전송 주기로 인하여 다시 연결된 MH와 오랜 시간 동안 전송할 수 없는 문제가 발생할 수 있다. 이를 극복하기 위하여 Freeze-TCP는 ZWP를 통한 연결 재개방식뿐만 아니라, 통신이 가능해졌을 때 MH로부터 이전에 수신한 마지막 세그먼트에 대한 세 개의 중복된 ACK를 보냄으로써 송신측에서 즉시 통신을 재개할 수 있도록 하였다. 즉, 연결이 재설정된 후 ZWP가 수신되지 않으면 MH는 연결해제 전에 마지막 패킷에 대한 ACK를 3개 복사해 연속으로 보낸다.

### 2.3 M-TCP

M-TCP는 무선링크에서 핸드오버 시 발생하는 빈번한 연결해제를 개선하고 패킷 손실을 줄이고자 제안된 알고리즘으로 유선링크에서는 표준 TCP를 사용하고 무선링크에서는 M-TCP를 사용한다[8]. M-TCP는 그림 3과 같이 SH(Supervisor Host), MSS(Mobile Support Station), MH의 세 개의 계층으로 구성된다.

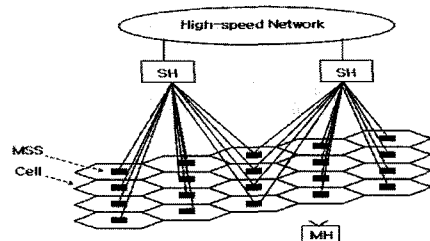


그림 3. M-TCP의 구조  
Fig 3. The Structure of M-TCP

M-TCP의 동작 과정은 FH가 MH에게 패킷을 전송하면 SH는 이 패킷을 M-TCP를 이용하여 MH에게 전달한다. 패킷을 수신한 MH는 SH에게 ACK를 보내며 SH는 ACK를 캐시에 저장한다. SH가 다음 ACK를 받으면 캐시에 저장된 ACK를 FH에게 전달한다. M-TCP의 핸드오버 처리 과정을 살펴보면 먼저 MH의 핸드오버를 감지한 SH가 저장되어 있는 마지막 ACK를 수신 윈도우 크기를 0으로 설정하여 FH에게 보내면 FH는 송신을 중단하고 핸드오버가 끝나면 SH는 송신 윈도우 크기를 조정하여 FH에게 보내어 전송을 재개한다. 핸드오버가 수행되는 동안에 FH가 데이터를 보내지 않도록 하여 데이터 손실을 막는다. 즉, M-TCP는 무선망에서 핸드오버로 인한 패킷 손실이나 전송 중단 현상이 일어나면, FH의 전송

윈도우 크기를 0으로 초기화하여 전송을 중단한다. 핸드 오버가 완료되면 핸드오버 발생 전의 크기로 전송을 시작함으로써 핸드오버 수행 동안의 전송된 데이터 손실과 불필요한 윈도우 크기의 감소를 방지함으로써 패킷 손실을 줄이고 TCP 성능을 향상시킨다.

2.4 SNOOP

SNOOP은 데이터 링크 계층에서의 재전송으로 상위 프로토콜과 독립적으로 안정된 재전송기반을 마련함으로써 무선채널의 높은 오류특성을 완화시키는 방식이다[10].

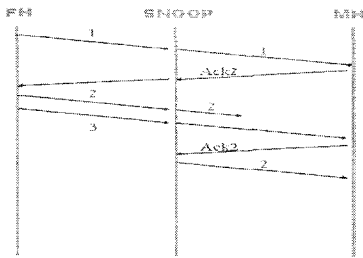


그림 4. SNOOP의 지역 재전송  
Fig 4. Local Re-transmission of SNOOP

SNOOP은 BS를 기준으로 유선연결과 무선연결을 구분하여 무선 환경의 패킷 손실에 대해 무선구간 내 지역 재전송을 통해 FH의 빠른 재전송 혹은 혼잡제어를 방지함으로써 TCP의 성능을 향상시킨다. SNOOP을 이용하기 위해서는 유선 구간의 종단인 BS의 네트워크 계층 소프트웨어를 변형시켜 전송받는 TCP 윈도우 크기의 버퍼를 갖는 SNOOP 모듈을 추가하여, TCP 세그먼트의 헤더를 분석할 수 있도록 한다.

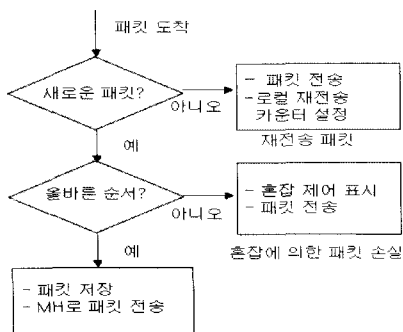


그림 5. BS가 패킷을 받았을 때의 처리과정  
Fig 5. The Process while the BS receive to packet

그림 5는 BS가 FH로부터 패킷을 받아서 처리하는 과정을 나타낸 것이다. SNOOP 모듈의 동작과정을 살펴보면 FH에서 패킷을 전송하면 SNOOP 모듈이 구현된 BS는 FH에서 들어오는 패킷들을 버퍼에 저장하고 MH로 전송한다. 그림 5는 새로운 패킷이 올바른 순서로 들어오면 패킷을 저장하고 MH로 전송하는 것을 보여준다. 만일 BS가 FH의 타임아웃이나 빠른 재전송에 의해 자신의 버퍼에 이미 저장되어 있는 패킷을 수신하면 BS는 이 패킷과 MH로부터 받은 마지막 ACK의 시퀀스 번호를 비교한다. ACK의 시퀀스 번호보다 패킷의 시퀀스 번호가 크면 MH는 해당 패킷을 전달받지 않았을 가능성이 크므로 MH에 해당 패킷을 전송한다. 만약 새로운 패킷인데도 올바른 순서가 아니라면 혼잡제어를 표시하고 패킷을 전송한다. 반대로 ACK 시퀀스 번호가 패킷 시퀀스 번호보다 크면 이미 MH가 수신한 패킷이므로 무시하고 FH로 ACK만 전송한다. BS가 받은 패킷이 버퍼에 저장되어 있지 않고 기다리던 패킷이 아닌 다른 시퀀스 번호의 패킷이라면 이는 유선망에서 혼잡에 의한 패킷 유실이거나 유선망에서 순서가 뒤바뀌어 전달된 경우이다. 이 경우 SNOOP 모듈에서 혼잡을 경험했다는 별도의 표시를 한 후 MH로 전송한다.

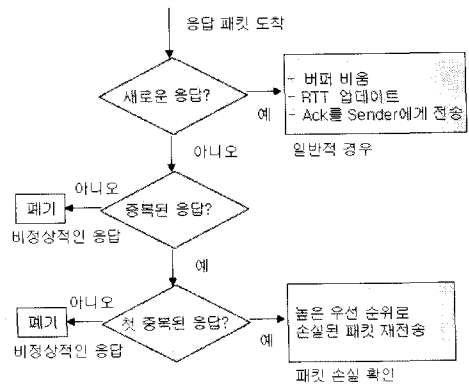


그림 6. BS가 ACK를 받았을 때의 처리과정  
Fig 6. The Process while the BS receive to ACK message

그림 6은 MH로부터 수신한 ACK에 대한 BS의 처리 과정을 나타낸다. BS가 MH에게 전달한 패킷들에 대해 ACK를 수신하면 BS는 해당 ACK를 송신측으로 전달한다. MH가 보낸 응답 ACK를 받았을 때, BS는 중복 응답 여부를 확인한 후 전송 혹은 폐기하는 것을 알 수 있다.

중복 ACK인 경우 중복 ACK의 시퀀스 번호를 보고 무선 구간에서 손실된 패킷이 버퍼에 저장되어 있는지 확인하고 저장되어 있으면 중복 ACK는 FH로 전달하지 않고 해당 패킷을 MH로 재전송한다. SNOOP은 무선 환경의 패킷 손실에 대해 무선 구간의 지역적 재전송을 통해 전체 망의 성능을 향상한다. 하지만 SNOOP은 연집오류에 대해서 오류 패킷에 대한 중복 ACK를 연속으로 받아야 하기 때문에 비효율적이므로 연집오류를 위하여 기존의 ACK(cumulative ACK) 대신 SACK(Selective ACK)를 사용한다. SACK는 최근 전송된 메시지 윈도우 안의 모든 패킷 시퀀스 번호를 비트 마스크(bit mask) 형태로 가지고 있기 때문에 이 정보를 해석하여 손실된 패킷만을 다시 받을 수 있다. RFC2018[11]는 SACK에서 ACK의 옵션 부분에 성공적으로 수신한 데이터 중에서 연속적으로 받은 4개의 블록까지 표기할 수 있다. SACK는 2개의 TCP 옵션과 초기 연결 설립 시 SYN 세그먼트를 포함하는 허용(permitted) 옵션, SACK의 활성화를 위한 SACK-옵션을 가지고 있다.

### III. 제안방법

#### 3.1 기존의 무선 TCP의 문제점

기존의 M-TCP는 잦은 단락 현상이 발생하거나 대역폭이 자주 변하는 무선망에 적합하지만, 낮은 비트 오류율을 가정하여 설계되었기 때문에 연집오류가 일어났을 경우 비효율적이다. 마찬가지로 SNOOP의 경우 단일 오류가 발생하면 지역적 재전송을 통해 TCP의 성능 향상에 기여하지만 연집오류가 발생할 경우 FH에서 재전송 타임아웃이 발생하여 혼잡모드로 동작하므로 FH로부터 불필요한 재전송이 일어나는 문제점을 가지고 있다. 그래서 연집오류를 위한 SACK 알고리즘을 이용하게 되는데 SACK는 전송된 메시지 윈도우 안에 모든 패킷들의 시퀀스 번호를 갖고 있기 때문에 손실된 패킷만을 다시 요구할 수 있다. 하지만 단일 전송 패킷들의 묶음인 메시지 윈도우 크기가 큰 경우 ACK의 생성과 전송에 오버헤드가 될 수 있다. 이러한 문제점을 해결하기 위해 RFC2018은 ACK 내에 기술할 수 있는 시퀀스 번호를 4개의 블록으로 제한하였다. 간헐적으로 발생하는 연속적 수신 데이터 블록이 4개 이상일 경우 해결할 수 없는 문제점을 갖는다.

#### 3.2 SACK-SNOOP의 개념

SACK-SNOOP은 TCP의 성능을 개선하기 위해 Freeze-TCP에 있는 ZWA 메시지를 이용하여 연집오류에 대하여 효율적으로 처리하는 프로토콜이다. 특히 SACK의 문제점인 간헐적으로 발생하는 연집오류를 해결하기 위해 SACK의 알고리즘을 개선하였다.

SACK에서 ACK는 누적 전송된 메시지 윈도우 안의 모든 패킷의 시퀀스 번호를 포함하기 때문에 이러한 비트 마스크 정보가 ACK의 생성과 전송에 오버헤드를 초래했다. 하지만 개선된 SACK 알고리즘은 전송된 모든 패킷의 시퀀스 번호를 포함하지 않고 오류 상황에 적합한 시퀀스 번호를 구성한다. 즉 연집오류의 개수에 따라 전송 받은 패킷이 오류 패킷보다 많은 경우에는 오류 패킷의 시퀀스 번호로 SACK의 ACK 옵션을 구성하고 전송 받은 패킷이 오류 패킷보다 적은 경우에는 전송 받은 패킷의 시퀀스 번호로 SACK의 ACK 옵션을 구성할 수 있도록 개선하였다. 이는 한 번에 전송되는 패킷들의 묶음인 메시지 윈도우 크기가 어느 정도 큰 경우에도 정확하게 연집오류의 패킷 시퀀스 번호를 전송할 수 있는 장점을 갖는다. 개선된 SACK는 무선망에서 단일오류의 경우 오류 패킷의 시퀀스 번호만 전송하고 연집오류의 경우 연집오류의 개수에 따라 전송 받은 혹은 오류가 발생한 패킷의 시퀀스 번호만을 전송할 수 있다. 단지 SACK-옵션은 SACK의 활성화를 위한 것이 아니라 패킷의 시퀀스 번호 혹은 오류가 발생한 패킷들의 시퀀스 번호를 가진 정보인지를 결정하는 플래그 정보로 변환되어야 한다. 개선된 SACK 알고리즘은 RFC2018에 나타난 최근 수신한 4개 블록의 데이터 시퀀스 번호뿐 아니라 최대 8개 블록의 데이터 시퀀스를 표시할 수 있기 때문에 곧바로 적용할 수 있는 알고리즘이다.

개선된 SACK 알고리즘을 통해 MH가 연집오류의 발생을 알리면 BS은 연집오류 패킷들을 바로 인식하여 재전송 모드로 동작한다. 재전송 요구를 받은 BS는 FH에게 ZWA 메시지를 보내어 FT(Freeze Time)시간 동안 전송지연을 한다. FT는 식 1과 같이 계산된다. 여기서 BM-RTT는 BS에서 MH까지의 RTT를 의미한다.

$$FT = BM-RTT * 손실패킷갯수 \text{ ---- (식 1)}$$

Freeze-TCP의 ZWA 메시지는 MH에서 FH로 전송되고 M-TCP의 ZWA 메시지는 분리된 라우터에서 FH로

전송된다. FH와 MH의 RTT는 일반적으로 이때 발생하는 지연시간을 의미한다. 하지만 FH가 멀수록 RTT는 무선링크상의 재전송과는 관계없이 길어지기 때문에 식 1과 같이 BS와 MH의 무선 링크상의 RTT만으로도 충분한 지연이 가능하다. ZWA 메시지를 받은 FH는 FT 시간 동안 윈도우 크기를 0으로 하여 전송을 중단하고 BS는 손실된 패킷들을 무선구간에서 재전송한다. 재전송이 끝나면 MH는 FH에게 ACK를 보내어 전송을 재개한다. 만일 FT 시간 보다 ACK가 더 빨리 FH에 도착한다면 FH는 ACK를 받은 순간 전송을 재개하여 망의 빠른 복구를 이루게 하지만 ACK가 오지 않은 상태에서 FT 시간이 미리 끝난다면 FH는 타임아웃이 되고 흐름제어 모드로 동작한다. 이와 같이 연접오류가 발생하여 손실 혹은 전송된 패킷들의 시퀀스 번호들을 일괄 처리함으로써 기존 SNOOP보다 연접오류에 대한 손실 패킷들을 더 빠르게 재전송을 할 수 있다. 그림 7은 SACK-SNOOP의 알고리즘을 나타낸 그림이다.

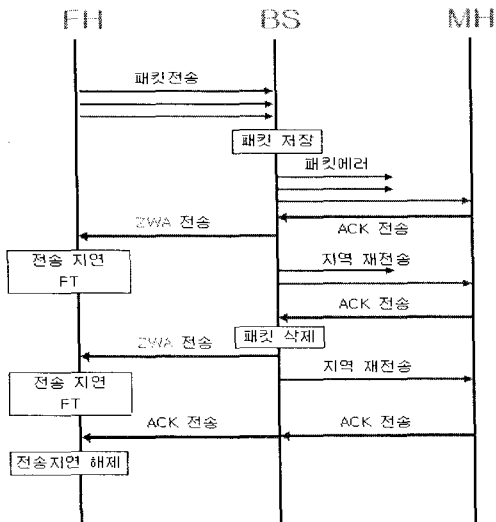


그림 7. SACK-SNOOP의 알고리즘  
Fig 7. The algorithm of SACK-SNOOP

3.3 단일오류의 경우

그림 8은 무선구간에서 1개의 패킷 오류에 대한 재전송 과정을 SNOOP과 비교한 그림이다. SNOOP은 중복 ACK를 사용한다. 무선 구간에서 2번 패킷 한 개가 손실되었다고 가정했을 때 기존 SNOOP은 무선구간에서 3번 패킷을 받은 후 2번 패킷을 받지 못했으므로 ACK2를

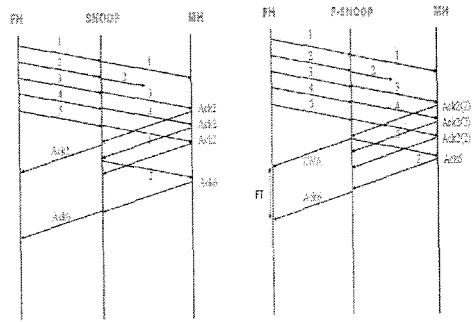


그림 8. 무선구간에서 단일오류  
Fig 8. Single error to Wireless network

보내고 4번 패킷을 받았을 때도 마찬가지로 ACK2를 보낸다. SNOOP은 이렇게 중복 ACK를 받으면 해당 패킷을 재전송한다. 그러나 SACK-SNOOP의 경우는 3번 패킷을 받았을 때 2번 패킷이 손실되었다는 것을 포함한 응답 Ack2(2)를 보낸다. 여기서 괄호안의 숫자는 손실된 패킷을 의미한다. 그러면 SACK-SNOOP에서 2번 패킷을 재전송함과 동시에 ACK2에 ZWA를 실어 보내어 FH에서 FT 시간 동안 패킷전송을 지연시키도록 한다. 지연 시간 안에 재전송이 끝나서 ACK가 전달되면 FH는 전송을 재개한다.

3.4 연접오류의 경우

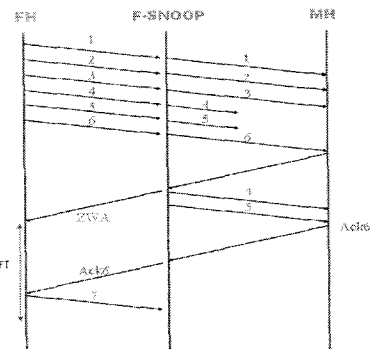


그림 9. 무선구간에서 연속 연접오류  
Fig 9. Bust errors to Wireless network

그림 9는 메시지 윈도우 크기가 6개의 패킷 묶음일 때 무선구간에서 오류가 2개 발생한 경우의 처리과정을 보여준다. SACK를 적용한 SNOOP은 무선구간에서 4, 5번 패킷의 손실이 발생하면 수신한 모든 패킷(1, 2, 3, 6번)

에 대한 정보를 비트 마스크 형태로 BS에 전송한다. BS는 이 정보를 해석하여 손실된 패킷을 찾아내어 MH에 재전송한다. 반면 제안한 SACK-SNOOP의 경우 MH는 손실된 패킷 정보인 비트 마스크를 포함한 SACK 메시지를 FH에게 보낸다. 그러면 BS는 이를 받고 4, 5번 패킷을 MH로 재전송함과 동시에 ACK에 ZWA 메시지를 실어 보내어 FH에서 FT 시간 동안 패킷 전송을 지연시킨다. 즉, 연접오류가 발생한 패킷의 개수와 비례하여 FH를 지연시키고 FT 시간 안에 ACK 메시지를 받으면 FH는 다시 정상적인 윈도우 크기로 패킷을 전송한다. 하지만 FT 시간 동안 ACK 메시지를 받지 못하면 FH는 흐름 제어 모드로 동작한다.

그림 10은 무선구간에서 4개 이상의 패킷이 연속적으로 오류가 발생했을 경우의 처리 과정이다.

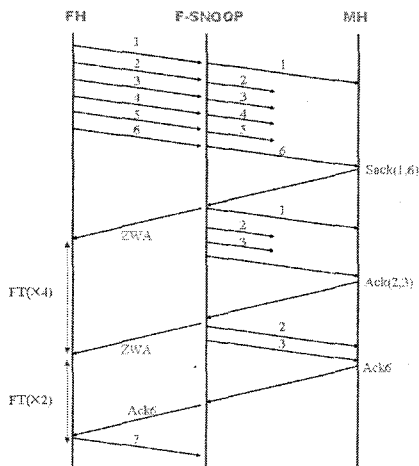


그림 10. 무선구간에서 재전송 오류  
Fig 10. The Re-transmission error to Wireless network

이 경우 연접오류의 패킷 개수가 전송 받은 패킷의 개수보다 크기 때문에 SACK를 이용한 SNOOP과 SACK-SNOOP은 거의 비슷하게 동작한다. 다만 SACK-SNOOP은 무선구간의 타임아웃을 없애기 위해 FT 시간 동안 FH의 전송을 지연시켜 유선구간에서 흐름제어 모드가 발생할 확률을 줄인다. 또한 BS에서 MH로 재전송할 때 발생할 수 있는 오류에 대해서도 FT 시간을 연장할 수 있다. 그러나 계속되는 무선구간의 오류는 흐름제어를 통한 해결 방법밖에 없기 때문에 재전송 오류에 대한 FT 시간의 연장은 연속적으로 적용하지 않고 단 한번만 적용하도록 한다. 이처럼 연접오류에 따라오는 재전송 오류

에 대해서 SNOOP은 지속된(persist) 타임아웃이 많아지고 FH에서 혼잡제어 모드로 동작하여 망의 효율을 떨어뜨릴 수 있다. 반면에 SACK-SNOOP은 무선구간의 재전송을 보다 효율적으로 이용함으로써 망의 이용율을 높일 수 있다.

#### IV. 실험 및 결과

##### 4.1 실험 환경

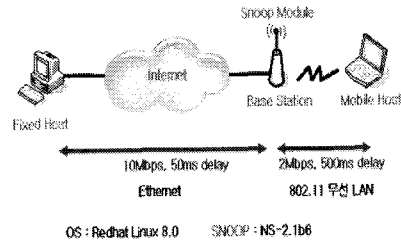


그림 11. 시뮬레이션 네트워크 구성도  
Fig 11. Network Structure of Simulation

SACK-SNOOP의 성능을 SNOOP과 비교하고자 미국 버클리 대학에서 개발한 NS-2(Network Simulator)[12]를 이용하였다. 시뮬레이션은 Redhat Linux 8.0 NS-allinone 2.1b6을 설치하여 실험했다. 그림 10은 시뮬레이션에서 사용된 네트워크의 구조와 링크 대역폭과 링크 지연시간을 보이고 있다. 유선은 10Mbps로 FH부터 BS까지 연결하였고 무선은 2Mbps로 BS에서 MH까지 연결하였다. 유선 링크 지연은 50ms, BS부터 MH까지의 무선 링크 지연은 500ms로 설정하였다. 실험에서 이용되는 TCP 트래픽의 파라미터는 표 1과 같다.

표 1. 실험을 위한 TCP 파라미터  
Table 1. TCP parameters for Simulation

파라미터	값
TCP Initial RTO	2.0(s)
TCP Maximum RTO	64(s)
TCP Minimum RTO	0.5(s)
TCP MSS	536 Bytes
TCP Maximum ACK Delay	0.01(s)
Simulation Time	80(s)

본 실험은 유선망의 오류는 고려하지 않고 무선망의 오류만을 가정한다. 무선망의 연접오류 특성을 모델링하기 위해 일반적으로 많이 사용되는 two-state 마코브

모델을 사용하여 무선 채널의 특성을 모델링하였다.

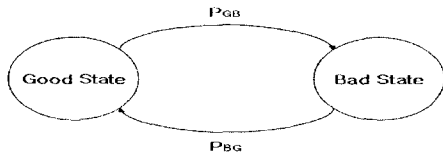


그림 12. two-state 마코브 모델  
Fig 12. two-state Markov model

two-state 마코브 모델은 양호 혹은 불량 상태에 대하여 각각  $10^{-6}$ ,  $10^{-2}$  의 BER을 갖는다. 양호 상태 체류 시간은 차수 4, 평균 1초의 Erlang 분포를 따르고 불량 상태 체류 시간은 차수 4, 평균 x ms의 Erlang 분포를 따르며 x는 2배씩 증가시켜가며 실험하였다. 표2는 MBP (Mean Bad Period)에 따른 무선링크의 평균 BER이다.

표 2. MBP에 따른 평균 BER  
Table 2. The average BER according to MBP

MBP	평균 BER
8 ms	$8.0 \times E-05$
16 ms	$1.58 \times E-04$
32 ms	$3.11 \times E-04$
64 ms	$6.02 \times E-04$
128 ms	$1.136 \times E-03$
256 ms	$2.039 \times E-03$

#### 4.2 실험 결과

그림 13은 TCP Reno, SNOOP, SACK-SNOOP을 적용하여 전송 동안의 TCP 윈도우 크기의 변화를 실험 한 결과이다.

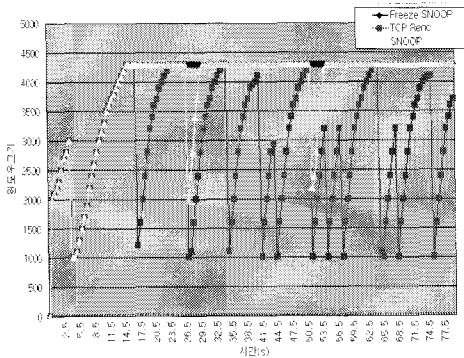


그림 13. TCP 윈도우 크기 변화  
Fig 13. The variation of TCP window size

TCP Reno의 경우 시간에 따라 윈도우 크기가 반복해서 증감하는 것을 관찰할 수 있는데, 이는 TCP가 혼잡을 제어하기 위해 윈도우 크기를 유동적으로 변화시키기 때문이다. SNOOP은 무선구간이 가지는 높은 오류율로 인한 연접오류가 발생한 부분들을 관찰할 수 있다. 이 부분에서 혼잡회피 단계로 진입하여 지역 재전송을 함으로써 TCP 윈도우 크기가 일정하게 안정되고 있다. 제안 방식은 ZWA 메시지를 FH에 보내 FT 시간 동안 윈도우 크기를 고정하고 지역 재전송을 하여 혼잡제어모드로 동작하는 것을 방지함으로 연접오류 부분에서 SNOOP 보다 윈도우 크기가 일정하게 안정적인 것을 보여준다.

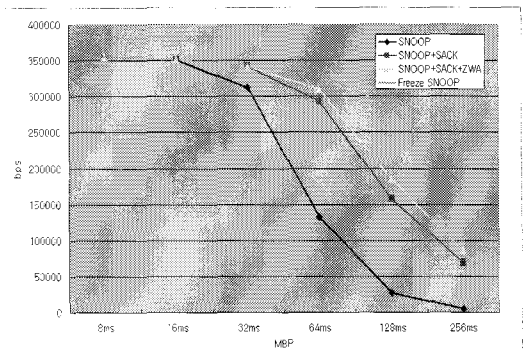


그림 14. MBP에 대한 전송효율의 비교  
Fig 14. The comparison of transmission rate according to MBP

그림 14는 MBP에 대한 전송효율을 SNOOP, SACK를 적용한 SNOOP, ZWA 메시지를 적용한 SNOOP, SACK-SNOOP에 대하여 비교하여 실험 한 그래프이다. 정확한 비교를 위해 각각의 오류 상황 중에서 MBP가 64, 128, 256ms인 경우 재전송시의 오류는  $6.02 \times E-04$ 로 고정하여 재전송 때에도 오류를 유발시켰다. 실험 결과에서 MBP가 8, 16, 32ms인 경우 SNOOP을 제외한 모든 알고리즘은 비슷한 전송효율을 보였고 재전송 오류를 적용한 64ms부터 제안 방식과 차이를 보였다. 그 중 SACK를 적용한 SNOOP과 ZWA 메시지를 적용한 SNOOP은 각각 미세한 차이를 보였다. 그 이유는 FH에서 흐름제어 모드로 동작할 때 발생하는 차이 때문이다. 제안 방식이 SACK와 ZWA 메시지를 적용한 SNOOP보다 더 좋은 전송효율을 가지는 이유는 개선된 SACK의 사용과 FT의 짧은 지연시간, 연접오류 환경에서 재전송 오류시의 FH에서의 FT시간을 재연장할 수 있기 때문이다. SACK-



SNOOP은 SACK를 적용한 SNOOP에 비해 평균 약 14.8%, ZWA 메시지를 적용한 SNOOP에 비해 평균 약 11.5%의 전송효율이 향상되었음을 확인하였다.

### V. 결 론

유무선 혼합망에서 TCP는 무선링크에서 무선망의 특징으로 인한 패킷 손실을 혼잡으로 인식한다. 이로 인하여 발생한 혼잡제어가 망의 성능저하를 초래한다. 이를 개선하기 위해 개발된 다양한 무선 TCP 방법중 SNOOP은 무선구간에서 지역적 재전송을 통해 FH에서의 빠른 재전송이나 혼잡제어를 일으키지 않도록 하여 TCP의 성능을 향상시킨다. 그리고 SNOOP은 무선구간에서 연집오류를 처리하는데 있어서 SACK를 이용하는 데 메시지 윈도우 크기가 너무 큰 경우에는 ACK의 생성과 전송에 오버헤드가 될 수 있다.

본 논문에서는 SACK를 개선하여 오류환경에 따라 전송되는 패킷의 시퀀스 번호를 줄여 ACK의 생성과 전송, 그리고 해석을 빠르게 처리되도록 하였고 Freeze-TCP의 ZWA 메시지를 이용해서 새로운 FT 시간을 정의하여 FH에서 타임아웃이나 혼잡제어 모드로 동작하는 것을 지연하고 재전송 오류 시 FT 시간을 재연장함으로써 무선구간에서 연집오류 패킷들의 재전송하는 시간을 보호하기 위한 방법으로 SACK-SNOOP을 제안하였다.

실험은 버클리 대학에서 개발한 NS-2를 이용하였고 무선망의 오류를 적용하기 위해 two-state 마코브 체인을 이용하여 실험하였다. 전송 동안의 윈도우 크기 변화와 MBP에 따른 전송효율을 비교하여 기존의 SNOOP에 비해 SACK-SNOOP의 성능이 향상되었음을 검증하였다.

향후 연구 과제로 유무선 혼합망에서 무선 TCP를 사용할 경우 무선상의 오류 복구뿐만 아니라 유무선상의 오류를 더 빨리 구분하여 데이터 손실에 대해 정확하고 빠른 대응을 하기 위한 방안이 연구되어야 할 것이다.

### 참고문헌

- [ 1 ] V. Jacobson, "Congestion Avoidance and Control", In Proceedings of ACM SIGcomm'88, Vol. 18, No. 4, p314-329, August 1988
- [ 2 ] H. Balakrishnan, "Challenges to reliable Data Transport over Heterogeneous Wireless networks", PhD Thesis, University of California at Berkeley, 1998.
- [ 3 ] G. Xylomenos, et al., "TCP performance issues over wireless links", IEEE Commun. Mag., p2-14, Fourth quarter 2000
- [ 4 ] Floyd, S., and Henderson, "The NewReno Modification to TCP's Fast Recovery Algorithm", Internet RFC 2582, Experimental, April 1999.
- [ 5 ] W. R. Stevens, "TCP Slow Start, Congestion Avoidance, Fast Retransmission, and Fast Recovery Algorithms", IETF, RFC 2001, Jan. 1997
- [ 6 ] Ren Wang, Massimo Valla, M. Y. Sanadidi, and Mario Gerla, "Adaptive Bandwidth Share Estimation in TCP Westwood", In Proc. IEEE Globecom 2002, Taipei, Taiwan, R.O.C., November 17-21, 2002
- [ 7 ] A. Bakre, B. R. Badrinath, "I-TCP : Indirect TCP for Mobile Hosts", Preceedings of the 15th International conference of distributed Computing systems, June 1995
- [ 8 ] K. Brown and S, Sigh, "M-TCP : TCP for mobile cellular networks", ACM computer communication review, vol. 1 27, no. 5, October 1997
- [ 9 ] Tom Goff, James Moronski, Vipul Gupta, "Freeze-TCP: A true end-to-end TCP enhancement mechanism for mobile environments" 1995
- [ 10 ] H. Balakishnan, S. Seshan, and R, H. Kartz, "Improving reliable transport and handoff performance in cellular wireless networks", ACM Wireless Networks, vol. 1, December 1995
- [ 11 ] S. Floyd, "TCP Selective Acknowledgment Options", RFC 2018, October 1996
- [ 12 ] The Network simulator (NS-2) <http://www.isi.edu/nsnam/ns/>

저자소개



안 치 현(Chi-hyun Ahn)

1998년 한남대학교 전자공학과 학사졸업  
2000년 경희대학교 전자공학과 석사졸업  
2000년~현재 경희대학교 전자공학과  
박사과정

2005년~현재 열린사이버대학교 정보통신학과 초빙교수  
※관심분야: Mobile IP, Wireless TCP, 이동 통신



김 형 철(Hyung-Chul Kim)

1990년 경희대학교 전자공학과 학사  
졸업  
1992년 경희대학교 전자공학과 석사  
졸업

2000년 경희대학교 전자공학과 박사 졸업  
2002년~현재 열린사이버대학교 정보통신학과 교수  
※관심분야: 패턴인식, 영상통신, 컴퓨터네트워크



우 중 정(Jongjung Woo)

1976. 3 ~ 1982. 8 경북대학교 전자공학  
과 학사  
1982. 8 ~ 1988. 8 산업연구원 책임연구원

1988. 8 ~ 1993. 5: 텍사스주립 대학(오스틴) 전기컴퓨터  
공학과 석사 및 박사  
1998. 2 ~ 1999. 2: 텍사스주립 대학(오스틴) 전기컴퓨터  
공학과 객원교수  
1993. 9 ~ 현재 성신여자대학교 교수  
※관심분야: 임베디드시스템, 모바일컴퓨팅, 컴퓨터  
구조, 병렬처리, 원격교육

김 장 형(Jang Hyung Kim)

현재 제주대학교 통신컴퓨터공학부 교수  
※관심분야: 영상처리, 인터넷컴퓨팅



이 대 영(Dae-Young Lee)

1964년 서울대학교 물리학과 학사 졸업  
1971년 캘리포니아주립대학교  
컴퓨터학과 석사 졸업

1979년 연세대학교 전자공학과 박사 졸업  
1999년 한국통신학회 회장  
1971년~현재 경희대학교 전자정보대학 교수  
※관심분야: 컴퓨터 네트워크, 디지털 워터마크, 지능  
형 홈네트워크



전 계 석(Kye-suk Jun)

1969년 연세대학교 전자공학과 학사  
졸업  
1973년 연세대학교 전자공학과 석사  
졸업

1983년 연세대학교 전자공학과 박사 졸업  
2002년 한국통신학회 회장  
1979년~현재 경희대학교 전자정보대학 교수  
※관심분야: 마이크로파 소자설계, 초음파영상시스  
템, 센서네트워크