무선 망에서의 TCP 성능 향상을 위한 제한적인 Indirect-ACK

(Limited Indirect Acknowledgement for TCP Performance Enhancement over Wireless Networks)

김 윤 주 † 이 미 정 †† 안 재 영 †††
(Yun-Joo Kim) (Meejeong Lee) (Jae-Young Ahn)

요 약 TCP는 전송 비트 오류에 의한 패킷 손실 확률이 매우 낮은 유선 망을 대상으로 설계된 프로토콜이므로, 이를 그대로 유·무선 통합 환경에 적용할 경우 TCP 송신단이 무선 망에서 발생한 전송 비트 오류로 인한 패킷 손실도 네트워크 혼잡에 의한 것으로 가정하여 송신단 전송률을 낮추기 때문에 성능이 저하하게 된다. 이에 본 연구에서는 일시적으로는 TCP의 종단간 연결이 분리되지만 궁극적으로는 종단간 연결 개념을 유지하면서, 종단간 연결 개념을 항상 유지하는 기존의 Snoop에 비해서 무선 네트워크에서 발생한 패킷 손실을 송신단에게 감추는 능력을 향상시키는 방안들을 제안하였다. 제안하는 방안들은 송신단에서 무선 네트워크에서의 패킷 손실을 발견할 염려가 있다고 판단되는 경우에 한하여 기지국에서 Indirect-ACK (Indirect Acknowledgement)을 발생하도록 하는데, 그 판단 기준에 따라 세 가지로 구분된다 한편, 제안하는 방안들은 기지국 버퍼에 임계치를 두어 기지국 버퍼의 길이가 이 임계치 이하일 때에만 Indirect-ACK을 발생할 수 있도록 함으로써 Indirect-ACK 사용으로 인한 기지국의 버퍼 오버헤드를 제한한다. 시뮬레이션을 통해 제안한 방안들의 성능 및 오버헤드를 비교하였고, 제안한 방안들이 제한된 용량의 버퍼로 기존의 Snoop에 비해 TCP 성능을 향상시킬 수 있음을 확인하였다.

키워드 : TCP, Snoop, I-TCP, Indirect-ACK, 무선 네트워크

Abstract With the original Transmission Control Protocol(TCP) design, which is particularly targeted at the wired networks, a packet loss is assumed to be caused by the network congestion. In the wireless environment where the chances to lose packets due to transmission bit errors are not negligible, though, this assumption may result in unnecessary TCP performance degradation. In this paper, we propose three schemes that improve the ability to conceal the packet losses in the wireless network while limiting the degree of violating TCP end-to-end semantics to a temporary incidents. If there happens a packet loss at the wireless link and there is a chance that the loss is noticed by the sending TCP, the proposed schemes send an indirect acknowledgement. Each of the proposed schemes uses different criteria to decide whether there is a chance that the packet loss occurred in the wireless part is noticed by the sender. In order to limit the buffer overhead in the base, the indirect acknowledgements are issued only when the length of buffer is less than a certain threshold. We use simulation to compare the overhead and the performance of the proposed schemes, and to show that the proposed schemes improve the TCP performance compared to Snoop with a limited amount of buffer at the base station.

Key words : TCP, Snoop, I-TCP, Indirect-ACK, wireless network

† 비 회 원 : 이화여자대학교 컴퓨터학과
yunjoo@etri.re.kr
†† 정 회 원 : 이화여자대학교 컴퓨터학과 교수
lmj@ewha.ac.kr
††† 비 회 원 : 한국전자통신연구원 무선방송연구소 연구원
jyahn@etri.re.kr
논문접수 : 2002년 2월 6일
심사완료 : 2003년 1월 21일

1. 서 론

인터넷에서 가장 널리 사용되는 트랜스포트 프로토콜인 TCP는 송신단에서 일정 시간이 지나도록 ACK(acknowledgement) 패킷이 도착하지 않아 타임아웃이 발생하거나, 데이타 패킷의 손실을 의미하는 중복 ACK 패킷

을 받으면 네트워크에 혼잡이 일어났다고 가정하여 전송 속도를 줄인다[1]. 이와 같은 가정은 일반적으로 전송 비트 오류에 의한 패킷 손실률이 극히 낮은 유선 망의 경우 매우 적합하다. 그러나, 최근 들어 수요가 증가하고 있는 무선 망의 경우에는 페이딩(fading), 잡음, 간섭 등에 의해 링크의 전송 비트 오류율이 상대적으로 훨씬 높고 결과적으로 이에 의한 패킷 손실이 자주 발생하는 특성을 지니고 있다. 따라서, 모든 패킷 손실에 대하여 혼잡 제어에 들어가는 TCP 프로토콜을 무선 망에 그대로 사용할 경우, 불필요하게 전송 속도가 감소되어 유선 망 활용이 저조해지고 처리율(throughput)이 저하하게 된다.

이에 무선 망에서의 TCP 성능 향상을 위한 여러 가지 개선안들이 제안되었는데[2,3,4,5,6,7,8,9,10,11], 이들 가운데 가장 대표적인 프로토콜로서 I-TCP와 Snoop 프로토콜을 들 수 있다. I-TCP는 기지국을 중심으로 TCP 연결을 분리하여 종단 간 연결을 유선 망과 무선 망의 두 부분으로 나누고 기지국에서 각각의 연결을 독립적으로 제어함으로써 무선 망에서의 오류를 송신단 측에 감추도록 하는 프로토콜이다[2,3,4]. I-TCP가 적용된 기지국은 기지국과 고정 송신단, 그리고 기지국과 이동 수신단 사이에 각각 I-TCP 소켓을 하나씩 생성하여 송신단과 수신단 사이의 I-TCP 연결을 설정하고, 기지국은 송신단에게 기지국이 수신단인 것처럼 수신단의 이미지를 갖고 통신을 시작하게 된다. 즉, 송신단이 보낸 데이타를 기지국이 수신하면 기지국은 일단 이에 대한 ACK을 송신단으로 전송하고, 기지국으로부터 수신단까지의 전송은 기지국에서 별도로 수행한다. 따라서, I-TCP의 경우는 무선 망에서 발생한 패킷 손실을 송신단에게 완전히 감출 수 있다. 그러나, 이 기법은 TCP의 종단 연결 개념에 위배되고, I-TCP가 적용된 수신단과 기지국의 전송 계층(Transport Layer)에 별도의 소프트웨어가 필요하다는 오버헤드가 있다. 또한, I-TCP의 성능은 기지국에 가용한 버퍼 크기에 따라 크게 달라지게 된다[12].

I-TCP와 달리 Snoop 프로토콜에서는 기지국이 기본적으로 데이타와 ACK을 릴레이하는 역할만을 담당하며, 무선 네트워크에서의 패킷 손실을 감추기 위해서는 중복 ACK을 스누핑하는 역할과 무선 링크 상에서의 오류에 의한 패킷 손실을 재전송에 의해 복구하는 역할만을 한다[5,6,7]. 따라서, TCP의 종단간 연결 개념을 유지하며, 기지국에서의 버퍼요구량도 최소이다. 그러나 기지국에서 지역적인 재전송을 통해 무선 링크에서의 패킷 손실을 복구하고 있는 동안 유선 망에 접속되어 있는 송신단은 ACK 패킷을 받지 못하기 때문에 유선 망에서의 가용 네트워크 대역폭 활용이 저하하게 될 수 있다. 더구나, 반복적 패킷 손실로 인해 지역적 재전송에 의한 패킷 손실 복구 작업이 오랫동안 지연되는 경우에는 무선 망에서 발생하는 패킷 손실을 송신단에게 숨길 수 없는 경우가 발생하게 된다.

이에 본 논문에서는 Snoop 프로토콜을 기반으로 하되 필요한 경우에만 제한적으로 I-TCP와 같이 기지국에서 ACK을 발생시킴으로써(이제부터 이를 Indirect-ACK이라 함), 유선 인터넷에 접속되어 있는 송신단과 무선 랜에 접속되어 있는 수신단이 TCP에 의하여 데이타를 전송할 때 이를 좀 더 효과적으로 지원할 수 있는 방안을 제안한다. 제안하는 방안들은 기존의 Snoop에 제한적인 Indirect-ACK 생성 기능을 추가하였는데, Indirect-ACK 사용으로 인하여 일시적으로는 TCP 연결을 분리하지만 궁극적으로는 TCP의 종단간 연결 개념을 유지하며 기지국에서의 버퍼 사용량을 제한하면서도 Snoop 보다 무선 링크에서의 패킷 손실을 더 효과적으로 감추고 처리율을 증가시킬 수 있다. 제안하는 방안들은 무선 링크에서 패킷 손실이 발생한 경우, 송신단에서 이를 발견할 염려가 있다고 판단되는 경우에만 Indirect-ACK을 발생시키되, 버퍼 사용량을 제한하고 종단간 연결 개념 복구에 걸리는 시간을 제한하기 위하여 버퍼 크기가 특정 임계치 이하인 경우에만 Indirect-ACK을 발생시킨다. 본 논문은 송신단에서 무선 링크에서의 패킷 손실을 발견할 염려가 있다고 판단하는 기준에 따라 세 가지의 서로 다른 스킴을 제안하고 이들의 성능 및 오버헤드를 비교하였다.

역시 유·무선망으로 구성되는 TCP 연결의 전송 효율을 향상시키기 위해 제안된 또 다른 프로토콜로서 W-TCP가 있다[8,9]. 제안하는 방안은 I-TCP, Snoop 등과 마찬가지로 무선 망에서의 전송 비트 오류로 인한 패킷 손실로 송신단의 전송률이 불필요하게 낮아지는 것을 방지하는 것을 목적으로 하는 반면, W-TCP는 무선 망 패킷 손실 복구에 걸리는 시간으로 인해 송신단의 네트워크 왕복시간(round trip time) 측정이 부정확하게 길어지고 이에 따라 송신단 재전송 타임 아웃이 길어져 유선 망에서의 혼잡 현상을 잘 파악하지 못하는 것을 방지하는 것을 목적으로 한다. 이 밖에도, 이동 무선 망 환경에서의 이동성에 의한 오류 및 전송 중단에 효율적으로 대처하기 위한 목적으로 M-TCP와 TCP-SMART가 제안된 바 있다. 제안하는 방안은 M-TCP나 TCP-SMART 등에서 다루고 있는 불연속적 연결로 인한 오류보다는, 상대적으로 취약한 무선 링크의 전송 비트 오류에 의한 패킷 손실에 대처하는 것을 목적으로

한다.

본 논문은 다음과 같은 순서로 구성되어 있다. 1장의 서론에 이어, 2장에서는 제안하는 스킴이 기반하고 있는 Snoop 프로토콜의 동작과 그 문제점을 살펴본 뒤, 3장에서 제한적으로 Indirect-ACK을 발생시키는 세 가지의 제안하는 스킴에 대하여 설명한다. 4장에서는 시뮬레이션 수행 결과를 통해 그 성능을 분석하고 마지막으로 5장에서 결론을 맺는다.

2. Snoop

본 장에서는 유선 환경과 무선 환경이 서로 연결되어 있는 네트워크에서의 TCP 성능 개선을 위하여 현재 가장 주목 받고 있는 프로토콜[13]이며, 제안하는 방식이 기반으로 하고 있는 Snoop 프로토콜의 작동 방식을 설명한다[5,6,7].

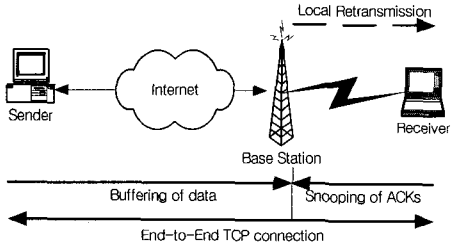


그림 1 Snoop 프로토콜의 시스템 구성도

그림 1에서 보이는 것과 같이 Snoop은 유선 망과 무선 망이 혼합된 환경에서 기지국에 설치되어 작동한다. Snoop은 각 패킷 흐름에 대해 버퍼를 할당하고 버퍼에 송신단으로부터 수신단으로 전송되는 데이타 패킷을 저장하며 송신단과 수신단 사이에 전송되는 ACK 패킷과 데이타 패킷을 각각의 목적지로 릴레이하는 기능을 기본적으로 구현한다. 그러나, 중복 ACK 패킷이 발생하는 경우에는 Snoop은 이것을 송신단에게 전달하지 않고, 기지국 버퍼에 저장되어 있는 패킷을 수신단에게 재전송(local retransmission)하여 패킷 손실을 지역적으로 복구함으로써 무선 망에서의 패킷 손실 발생을 송신단에게 감춘다. 또한, 버퍼의 가장 앞에 있는 패킷에 대해 패킷을 전송한 시간을 시작으로 해서 일정 시간 지속되는 지역 재전송 타이머를 설정해 두고, 이 지역 재전송 타이머가 만기되도록 패킷에 대한 ACK이 도착하지 않는 경우에도 무선 망 전송에서 패킷 혹은 ACK이 손실되었다고 가정하고 수신단에게 해당 패킷을 재전송한다.

데이타 패킷이 기지국의 Snoop 모듈을 통과하여 수신단에게 전달되는 과정은 그림 2(a)와 같다. 먼저 기지국에 데이타 패킷이 도착하게 되면, Snoop 모듈은 도착한 데이타 패킷이 새로운 패킷인가를 확인하고 새로운 패킷이 아니면 저장하지 않고 바로 수신단에 패킷을 전달한다. 만약 도착한 패킷이 새로운 패킷이지만 순서에 맞지 않으면 기지국은 유선 망에서 혼잡으로 인해 패킷 손실이 발생했다고 보고 해당 패킷에 대해 혼잡을 표기한 후 수신단으로 전송한다. 그리고 도착한 패킷이 순서에 맞게 올바르게 도착한 새로운 패킷이면 복사본을 기지국의 버퍼에 저장하고 수신단에게 전달한다.

그림 2(b)는 TCP 수신단으로부터 ACK 패킷을 수신하였을 때 기지국에서 동작하는 Snoop 모듈의 작동 과정을 보여준다. 도착한 ACK 패킷이 새로운 ACK 패킷이면 Snoop은 기지국 버퍼에 저장되어 있는 데이타 패킷들 중

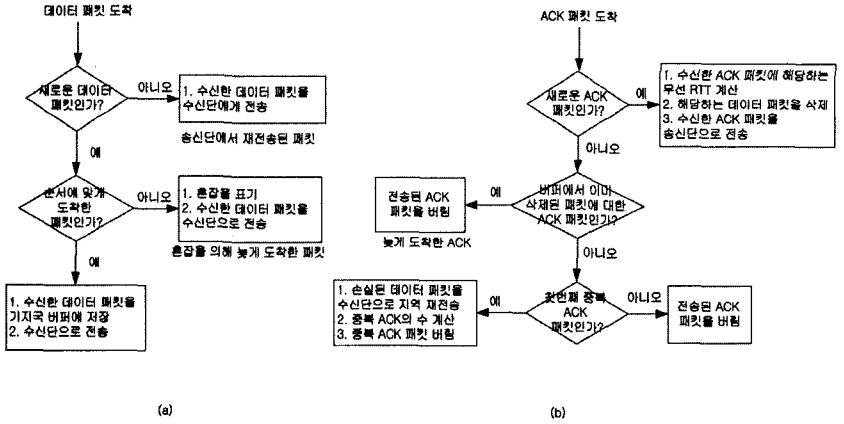


그림 2 Snoop의 패킷 처리 과정

해당 패킷들을 버퍼에서 삭제하고, 무선 링크의 RTT(Round Trip Time)를 측정하여 이 새로운 RTT를 반영하도록 지역 재전송 타이머 값을 수정한다. 도착한 ACK 패킷이 이미 버퍼에서 삭제된 패킷에 대한 ACK이라면 기지국은 수신한 ACK 패킷을 단순히 버린다. 만약 도착한 ACK 패킷이 첫번째 중복 ACK이라면 이것은 무선 망의 패킷 손실을 의미하므로 기지국 버퍼에 저장되어 있는 데이타 패킷들 중 손실된 데이타 패킷을 TCP 수신단에게 재전송하고, ACK 패킷은 버린다. 그러나 만약 중복 ACK 패킷이지만 첫번째 중복 ACK 패킷이 아니라면 이미 지역 재전송을 한 패킷이므로 단순히 ACK 패킷을 버린다.

이와 같이 동작하는 Snoop 프로토콜은 TCP의 종단간 연결 개념을 유지할 수 있다는 장점이 있으나, Snoop이 무선 망에서 발생한 패킷 손실을 지역 재전송에 의해 복구하는 동안에 TCP의 송신단은 ACK 패킷을 기다려야 하므로 송신단 휴지 시간이 발생하여 유선망의 자원을 낭비하게 된다. 또한 반복적인 패킷 손실로 인해 기지국에서의 지역적 패킷 손실 복구에 걸리는 시간이 지나치게 길어지는 경우에는 무선 망의 패킷 손실이 송신단에게 숨겨지지 못해 송신단에서 재전송 타임아웃과 불필요한 혼잡 제어가 시작될 가능성이 있다.

3. 제안하는 TCP 성능 향상 방안

본 연구에서는 기존의 Snoop에 비하여 좀 더 무선 망에서의 패킷 손실을 감출 수 있는 능력을 향상시키기 위하여 제한적으로 Indirect-ACK을 사용하도록 Snoop을 확장시키는 방안을 제안한다. 기존의 I-TCP 등에서 TCP 연결을 완전히 분리하여 기지국이 수신한 모든 패킷에 대하여 Indirect-ACK을 발생시키는 것과는 달리, 제안하는 Indirect-ACK 사용 방식에서는 기지국에서의 버퍼 사용량을 제한하고 연결 분리를 일시적인 것으로 제한하기 위하여 무선 링크에서 패킷 손실이 발생하여 송신단에서 이를 감지할 우려가 있다고 판단되는 경우에만 Indirect-ACK을 발생시키되, 기지국 버퍼 길이가 특정 임계치 미만인 경우에만 Indirect-ACK이 발생되도록 한다.

이 때, 버퍼 길이에 대한 임계치는 무선 링크에서 손실된 패킷이 복구되는 시간 동안 기지국과 수신단 사이에 성공적으로 전송될 수 있는 데이타 패킷의 수를 대중한 값으로 정하였으며, Snoop의 지역 재전송 타이머 값(local retransmission timeout)과 무선 링크의 왕복 전파 시간(wireless link RTT)을 이용하여 아래와 같이 계산하였다.

$$\text{버퍼임계치} = \frac{\text{local retransmission timeout} + \text{wirelesslinkRTT}}{\text{wireless link RTT}}$$

이와 같은 버퍼 임계치를 사용함으로써, 손실된 패킷의 재전송을 시작하기까지 가장 오랜 시간이 소요되는 경우인 기지국의 재전송 타이머의 만기에 의해 재전송이 시작되는 경우에 손실된 패킷이 한 번의 재전송에 의하여 복구되는 경우를 기준으로 하나의 패킷을 복구하는데 걸리는 시간동안 기지국이 수신단으로 내보낼 수 있는 양보다 기지국 버퍼 길이가 적은 경우에만 송신단에게 Indirect-ACK을 보내주게 된다.

본 논문에서는 무선 링크에서의 패킷 손실을 송신단에서 감지할 수 있는 가능성이 있다고 판단하는 기준 및 어느 패킷에 대하여 Indirect-ACK을 발생시키는가를 정하는 기준에 따라 제안하는 스킴을 WTS(Wireless Transmission Status), IAT-FP(Indirect ACK Timer set on the First Packet), IAT-LP(Indirect ACK Timer set on the Last Packet) 등의 세 가지로 분류한다. 첫 번째 스킴인 WTS에서는 무선 링크에서 패킷 손실이 발생한 것을 발견하게 된 경우 항상 송신원에서 이를 감지할 우려가 있다고 가정하고 손실된 패킷에 대한 ACK이 도착할 때까지 그 이후에 도착하게 되는 모든 패킷에 대하여 버퍼 길이 제약이 허용하는 한 Indirect-ACK을 발생시킨다. 나머지 두 스킴은 Indirect-ACK 타이머를 사용하여 Indirect-ACK 발생 시기와 Indirect-ACK으로 승인할 패킷을 결정하는데, IAT-FP에서는 기지국 버퍼에서 아직 송신단에 ACK을 보내지 않은 첫 번째 패킷에 대하여 Indirect-ACK 타이머를 두는 반면, 세 번째 스킴인 IAT-LP에서는 기지국 버퍼에 마지막으로 도착한 패킷에 대하여 Indirect-ACK 타이머를 둔다. Indirect-ACK 타이머에 기반한 이 두 스킴은 Indirect-ACK 타이머가 만기되면 Indirect-ACK 타이머가 설정되어 있는 패킷에 대하여 Indirect-ACK을 발생시킨다.

세 가지 스킴들 가운데 WTS가 가장 먼저 Indirect-ACK을 발생시키기 시작하고 다음으로 IAT-FP, 그리고 IAT-LP가 가장 늦게 Indirect-ACK을 발생시키기 시작한다. 또한, WTS는 일단 Indirect-ACK이 발생되기 시작하면 그 이후에 도착하는 매 패킷에 대해 Indirect-ACK을 발생시킬 수 있고, IAP-FP는 Indirect-ACK을 발생시키는 해당 패킷들의 도착 시간 간격마다 Indirect-ACK을 발생할 수 있는 반면, IAT-LP는 일반적으로 Indirect-ACK 타이머 간격이 지나야 다음 Indirect-ACK을 발생시킨다. 즉, IAT-LP는 WTS나 IAT-FP에 비하여 Indirect-ACK 발생 빈도가 낮다.

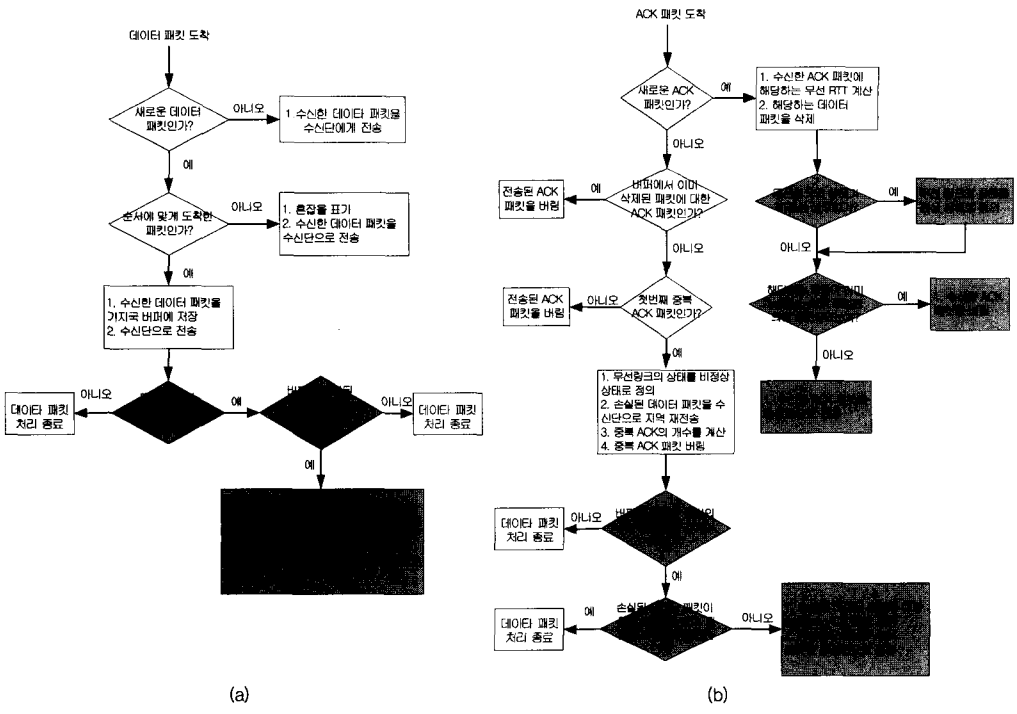


그림 3 WTS의 패킷 처리 과정

3.1 WTS(Wireless Transmission Status) 작동 방식

WTS는 중복 ACK을 수신하거나 기지국 버퍼에 설정되어 있는 지역 재전송 타이머가 만기되면(즉, 무선 망에서의 패킷 손실을 발견하면) 항상 무선 망의 패킷 손실이 송신단에서 감지될 가능성이 있다고 가정하고 수신단으로부터 손실된 패킷에 대한 ACK이 도착할 때까지 기지국 버퍼 길이가 버퍼 임계치보다 적은 한 기지국에 도착하는 모든 데이타 패킷들에 대하여 Indirect-ACK을 발생시킨다. 즉, WTS는 현재 버퍼의 가장 앞에 있는 패킷에 대하여 지역적인 재전송에 의한 손실 복구 작업 중인가의 여부에 따라, 무선 링크 상태를 비정상(복구 작업 중인 상태)과 정상 두 가지로 분류하고 정상 상태에서는 Snoop과 같이 동작하고 비정상 상태에서는 송신단으로부터 전송되는 패킷들에 대하여 Indirect-ACK을 발생시킨다.

WTS가 송신단으로부터 전송된 데이타 패킷과 수신단이 보내온 ACK을 처리하는 과정은 각각 그림 3(a), (b)와 같다. 순차 번호에 맞는 새로운 데이타 패킷이 기지국에 도착하면, 기지국은 수신한 데이타 패킷을 기지국 버퍼에 저장하고 수신단으로 전송한 후 현재 설정되어 있는 무선 링크의 상태를 판단한다. 만약 무선 링크의 상태가 비정상 상태라면 기지국은 버퍼의 길이가 임계치보다 작거나 같은 한 데이타 패킷을 수신할 때마다 지난 번에 전송한 Indirect-ACK 패킷에서 승인한 패킷의 다음 패킷에 대하여 Indirect-ACK을 생성한다.

만약 기지국에 새로운 ACK 패킷이 도착하면, 기지국은 수신한 ACK 패킷에 해당하는 데이타 패킷을 버퍼에서 삭제하고 해당 무선 링크의 RTT를 계산한다. 또한, 기지국의 상태를 정상 상태로 정의하고, 수신한 ACK 순차 번호가 아직까지 Indirect-ACK에 의하여 승인된 바 없는 순차 번호라면 수신한 ACK을 송신단으로 전송한다. 그러나, 수신한 ACK이 첫번째 중복 ACK이라면, 기지국은 무선 링크의 상태를 비정상 상태로 정의한 후, 수신한 중복 ACK에 해당하는 손실된 데이타 패킷을 지역적으로 재전송한다. 또한, 기지국 버퍼의 길이가 버퍼 임계치보다 작거나 같으면 손실된 패킷에 대한 Indirect-ACK 패킷을 생성하여 송신단으로 전송한다. WTS는 중복 ACK을 수신한 경우뿐만이 아니라 Snoop의 지역 재전송 타이머가 만기될 때까지 패킷에 대한 ACK이 도착하지 않으면 무선 링크의 상태를 비정상 상태로 정의하고, 버퍼의 길이가 버퍼 임계치보다 작거나 같으며 손실된 패킷이 아직 Indirect-ACK에 의해서 승인되지 않았다면 손실된 패킷에 대한

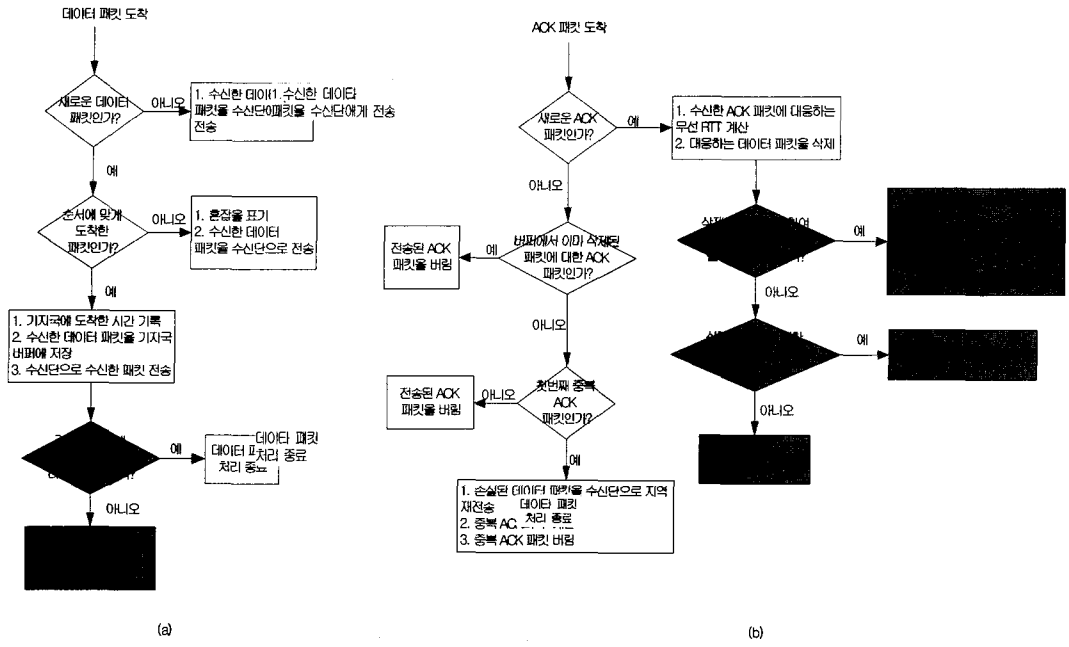


그림 4 IAT-FP의 패킷 처리 과정

Indirect-ACK을 생성하여 송신단으로 전송한다.

3.2 IAT-FP(Indirect ACK Timer set on the First Packet) 작동 방식

IAT-FP는 Snoop의 지역 재전송 타이머와 별도로 Indirect-ACK 타이머를 사용한다. Indirect-ACK 타이머는 기지국 버퍼에 있는 패킷들 가운데 아직 송신단에 승인하지 않은 패킷 중 가장 먼저 도착한 패킷에 대하여 그 패킷의 기지국 도착 시간에 Indirect-ACK 타이머 길이를 더한 값으로 설정된다. Indirect-ACK 타이머가 만기되면 IAT-FP는 해당 패킷에 대한 ACK 배달이 지나치게 지연되고 있다고 판단하고 Indirect-ACK을 발생시킨다. Indirect-ACK 타이머의 길이는 아래와 같이 계산되어 정해진다.

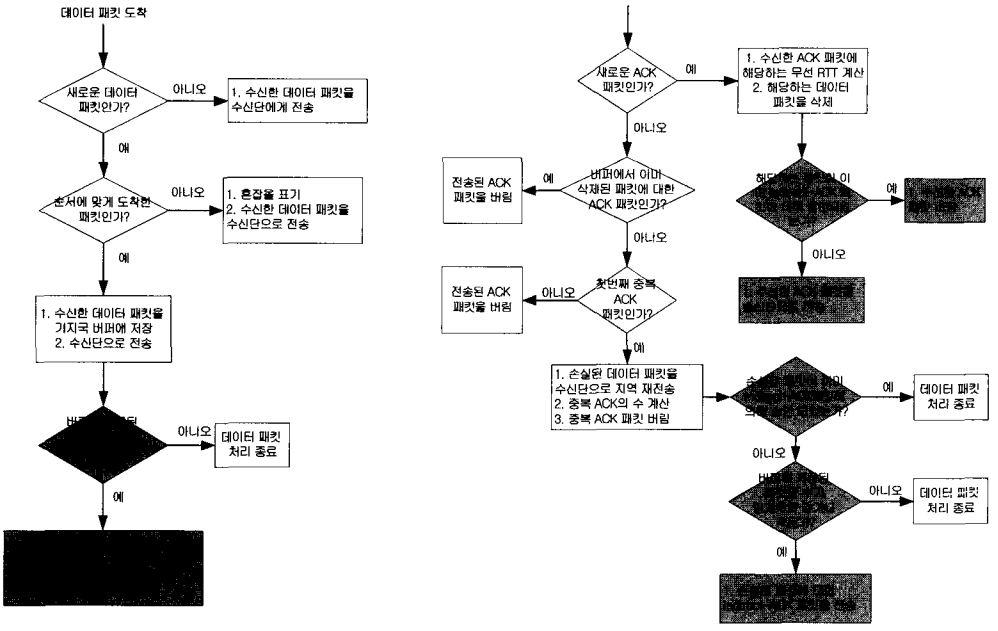
Indirect-ACK 타이머 길이

= Snoop의 지역 재전송 타이머 길이+무선 링크의 왕복전파시간

이것은 지역 재전송 타이머 만기로 손실이 확인되어 재전송된 패킷에 대한 ACK이 무선 링크 왕복 시간이 지나도 수신원으로부터 오지 않는 경우, 지역 재전송에 의한 패킷 손실 복구가 장기화되고 있다고 판단하는 것을 의미한다.

그림 4(a)는 IAT-FP가 데이타 패킷을 처리하는 과정에서 Indirect-ACK 타이머를 설정하는 과정을 도식화한 것이다. 기지국에 데이타 패킷이 도착하면 IAT-FP는 먼저 새로운 데이타 패킷인지 판단한다. 만약 새로운 데이타 패킷이라면 Snoop과 마찬가지로 이를 기지국 버퍼 저장한 후 송신단으로 전송한다. 또한, 이후 이 패킷에 대하여 Indirect-ACK 타이머를 설정해야 하는 경우를 위해 수신한 데이타 패킷이 기지국에 도착한 시각을 기록한다. IAT-FP는 이와 같이 무선 링크에서의 패킷 전송이 손실에 의해 지나치게 오랫동안 지연되고 있는가를 패킷 도착 시간으로부터 일정 기간(Indirect-ACK 타이머 길이) 내에 수신단으로부터 ACK이 오는가에 의하여 판단하기 때문에 판단하는 기준이 개념적으로 매우 직접적이고 단순하다는 장점이 있지만 이를 위해 각 패킷에 대하여 도착 시간을 기록해야 하는 오버헤드가 있다. 현재 기지국 버퍼에 Indirect-ACK 타이머가 설정되어 있는 패킷이 없는 경우에는 도착한 데이타 패킷에 대하여 Indirect-ACK 패킷을 설정한다.

그림 4(b)는 IAT-FP가 수신한 ACK 패킷을 처리하는 과정에서 Indirect-ACK 타이머를 설정하는 과정을 도식화 한 것이다. IAT-FP는 새로운 ACK 패킷을 수신하면 Snoop과 마찬가지로 수신한 ACK 패킷에 해당하는 데이타 패킷들을 기지국 버퍼에서 삭제한다. 그런데, 만약 삭제된 데이타 패킷 중 Indirect-ACK 타이머가 설정되어 있었던 패킷이 있다면 패킷 삭제후 기지국 버퍼의 가장 앞에 오게 되는 패킷에 대하여 다시



(a) (b)

그림 5 IAT-LP의 패킷 처리 과정

Indirect-ACK 타이머를 설정하고, 수신단으로부터 온 ACK을 송신단에 전달한다.

IAT-FP는 기지국 버퍼에 설정되어 있는 Indirect-ACK 타이머가 만기될 때까지 Indirect-ACK 타이머가 설정되어 있는 패킷에 대한 ACK 패킷이 도착하지 않으면 무선 링크에서의 패킷이 손실 시간이 길어져 송신단의 휴지 시간이 지나치게 길어지고 있거나 송신단에서 패킷 손실을 감지할 수 있다고 보고, 기지국 버퍼의 길이가 버퍼 임계치보다 작거나 같다면 Indirect-ACK 타이머가 설정되어 있는 데이타 패킷에 대한 Indirect-ACK 패킷을 송신단으로 전송한다. 그리고, Indirect-ACK을 발송한 패킷의 다음에 위치한 패킷에 대하여 그 패킷이 기지국에 도착한 시각을 기준으로 Indirect-ACK 타이머를 새로이 설정한다. 기지국 버퍼의 길이가 버퍼 임계치보다 크다면 Indirect-ACK 패킷을 발생하지 않고, 단지 현재 Indirect-ACK 타이머가 설정되어 있는 데이타 패킷의 다음에 위치한 데이타 패킷으로 Indirect-ACK 타이머 재설정만을 해 준다.

3.3 IAT-LP(Indirect ACK Timer set on the Last Packet) 작동 방식

IAT-LP는 IAT-FP와 마찬가지로 Snoop의 지역 재전송 타이머와 더불어 별도의 Indirect-ACK 타이머를 사용한다. IAT-LP의 Indirect-ACK 타이머는 새로 패킷이 도착할 때마다, 그 패킷의 도착시간에 Indirect-ACK 타이머의 길이를 더해 새로 설정된다. 따라서, IAT-LP의 경우에는 Indirect-ACK 타이머 설정을 위하여 각 패킷 별로 도착 시간을 기록하지 않아도 된다. 단, IAP-LP는 기지국 버퍼 길이가 버퍼 임계치를 초과한 경우에는 새로 패킷이 도착해도 Indirect-ACK 타이머를 재설정하지 않는다. IAT-LP는 Indirect-ACK 타이머의 만기가 발생하도록 수신단으로부터 그 패킷에 대한 ACK을 받지 못하면 해당 패킷에 대한 송신단으로의 ACK 배달이 지나치게 지연되고 있다고 보고 Indirect-ACK 타이머가 만기된 패킷에 대하여 Indirect-ACK을 발생한다. 또한, 중복 ACK을 수신하면 WTS와 마찬가지로 송신단에서 무선 링크에서의 패킷 손실을 감지할 가능성이 있다고 판단하고 아직 기지국 버퍼 길이가 버퍼 임계치보다 작거나 같다면 송신단에 Indirect-ACK을 보낸다.

그림 5(a)는 IAT-LP가 송신단으로부터 패킷을 받았을 때의 처리 과정을 도식화한 것이다. IAT-LP는 기지국에 새로운 데이타 패킷이 도착하면 역시 Snoop과 마찬가지로 먼저 순서에 맞는 새로운 패킷인지 확인하고 순서에 맞는 새로운 데이타 패킷이라면 데이타 패킷을 기지국 버퍼에 저장하고 수신단으로 전송한다. 또한, 이 패킷을 수신

하고도 버퍼 길이가 아직 버퍼 임계치보다 같거나 작다면 이전에 설정한 Indirect-ACK 타이머를 취소하고 새로 도착한 데이타 패킷에 대하여 Indirect-ACK 타이머를 다시 설정한다.

그림 5(b)는 기지국이 Indirect-ACK을 생성하는 과정을 도식화한 것이다. 기지국에 ACK 패킷이 도착하면 IAT-LP는 우선 새로운 ACK 패킷인지 확인한다. 만약 새로운 ACK 패킷이라면 Snoop과 마찬가지로 기지국 버퍼에서 해당하는 데이타 패킷들을 삭제한다. 또한, 수신단이 보낸 ACK 패킷이 승인하는 패킷 순차 번호에 대한 Indirect-ACK이 송신단에게 이미 전송된 바 있는지를 점검하여 아직 승인되지 않은 순차 번호가 있다면 수신단이 보낸 ACK 패킷을 송신단으로 전송한다.

그러나, 기지국에 도착한 ACK 패킷이 첫번째 중복 ACK 패킷이라면 IAT-LP는 Snoop과 마찬가지로 손실된 패킷을 재전송하고 수신된 중복 ACK 패킷은 버린다. 또한, 기지국 버퍼의 길이와 버퍼 임계치를 비교하여 버퍼 길이가 버퍼 임계치보다 작거나 같고, 손실된 패킷이 아직 Indirect-ACK에 의하여 승인된 바 없다면 이에 대한 Indirect-ACK을 송신단에 전송한다.

4. 성능 평가

제안하는 WTS(Indirect ACK Scheme based on Wireless Transmission States), IAT-FP(Indirect ACK Scheme based on Packet Arrival Time), IAT-LP (Indirect ACK Scheme with Indirect ACK timer)와 기존의 Snoop 프로토콜의 성능 비교를 위하여 시뮬레이션을 수행하였다. 먼저 4.1절에서 시뮬레이션에서 사용한 네트워크와 트래픽 모델에 대하여 설명하고, 4.2절에서는 시뮬레이션의 결과를 분석한다.

4.1 시뮬레이션 모델

시뮬레이션은 버클리 대학(U.C Berkeley)의 네트워크 시뮬레이터(Network Simulator)[14]를 이용하여 구현되었으며, 유닉스 워크스테이션에서 수행되었다. 그림 6은 본 시뮬레이션의 네트워크 모델과 링크 특성을 보여준다. 유선 링크는 10Mbps의 대역폭을 가지고 있으며 전송 지연은 100ms이다. 반면 무선 링크는 10ms의 전송 지연과 2Mbps의 대역폭 특성을 갖고 있다[7,12,15].

표 1은 시뮬레이션 모델의 주요한 특성들을 요약한 것이다. 송·수신단의 TCP는 Reno이며, 고정 망에 연결된 송신단은 FTP 혹은 TELNET 응용 프로그램에서 발생한 데이타를 무선 망에 연결된 수신단에게 전송한다. 무선 링크의 특성에 대한 TCP 성능을 평가하기 위

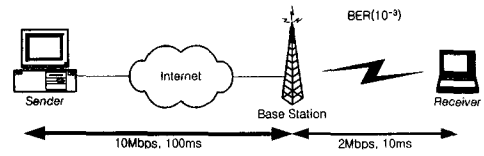


그림 6 시뮬레이션 네트워크 모델

하여 유선 링크에서의 혼잡 발생 확률은 0으로 설정하였다. 무선 링크의 비트 오류는 '좋은 상태'와 '나쁜 상태'의 두 가지 상태로 구성되는 Markov 에러 모델에 의해 발생하도록 하였는데, 이 때에 좋은 상태에서의 평균 비트 오류율은 10-6으로, 나쁜 상태에서의 평균 비트 오류율은 10-3으로 설정하였고, 좋은 상태가 지속되는 평균 기간은 0.8초, 나쁜 상태가 지속되는 평균 기간은 0.2초로 설정하였다[12,15].

표 1 시뮬레이션 모델의 주요 특성

Parameter	Value
Maximum packet size	1000bytes
Network data type	FTP / TELNET
Transport Protocol	TCP Reno
BER range	10-6~10-3
Error model	Two State Markov Chain
Mean good/bad period	0.8 sec / 0.2 sec

4.2 시뮬레이션 결과 및 분석

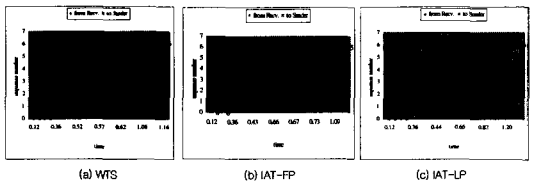
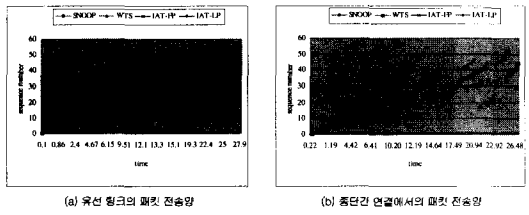


그림 7 제안하는 WTS, IAT-FP, IAT-LP의 Indirect-ACK 발생 과정 비교

그림 7은 세 가지 스킴의 Indirect-ACK 발생 과정 예를 보여 준 것이다. 그래프는 시간이 진행됨에 따라 기지국이 수신단으로부터 수신하게 되는 ACK 패킷의 순차 번호와 기지국에서 송신단으로 발송한 ACK 패킷의 순차 번호를 보여주고 있다. 그래프 상의 원 안에 있는 표기는 기지국에서 발생한 Indirect-ACK의 순차 번호이다.

WTS의 예는 패킷 2에 설정된 지역 재전송 타이머 종료로 인해 순차 번호 2를 갖는 Indirect-ACK이 송신단으로 전송되는 경우를 보여주고 있다. 무선 링크의 상

태를 비정상 상태로 정의한 WTS는 손실된 패킷 2가 복구되는 동안 기지국에 도착하는 데이타 패킷 3과 4에 대하여 모두 Indirect-ACK 패킷을 생성하여 송신단으로 전송한다. IAT-FP 예도 역시 패킷 2에 설정된 Indirect-ACK 타이머가 종료되어 패킷 2에 대한 Indirect-ACK이 생성된 경우를 보여주고 있다. 또한, 이 예에서는 패킷 2의 다음 패킷인 패킷 3이 패킷 2에 대한 Indirect-ACK이 발송된 이후 한참 뒤에 들어오고 패킷 4~6까지는 패킷 3의 뒤를 연이어 들어왔는데, 패킷 3에 대해서도 타임아웃이 발생할 때까지 수신단으로부터 패킷 3에 대한 ACK이 도착하지 않아 패킷 3에 대해서도 Indirect-ACK이 발생되었다. 그리고 패킷 3에 연이어 들어온 패킷 4, 5, 6에 대해서도 이 패킷들이 기지국에 도착한 시간 차이를 간격으로 Indirect-ACK 패킷이 발생되었다. IAT-LP에 대한 예에서도 데이타 패킷 2에 설정된 Indirect-ACK 타이머가 종료될 때까지 수신단으로부터 해당하는 ACK 패킷이 도착하지 않아 패킷 2에 대한 Indirect-ACK이 송신단에 발송되었는데, 손실된 패킷 2를 지역 재전송에 의해 복구하는 동안 기지국 버퍼에는 패킷 5까지가 도착하였고, 이 마지막 패킷 5에 대해 Indirect-ACK 타이머가 설정되었다. 그러나 이 패킷 5에 대한 Indirect-ACK 타이머가 종료될 때까지 해당하는 ACK 패킷이 도착하지 않아 패킷 2에 이어 패킷 5에 대한 Indirect-ACK이 송신단으로 전송되었다.

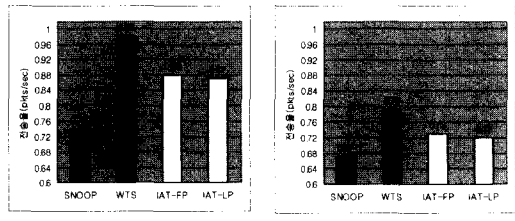


(a) 유선 링크의 패킷 전송양 (b) 종단간 연결에서의 패킷 전송양

그림 8 FTP 전송시 시간 변화에 따른 WTS, IAT-LP, IAT-FP와 Snoop의 데이타 전송양 비교

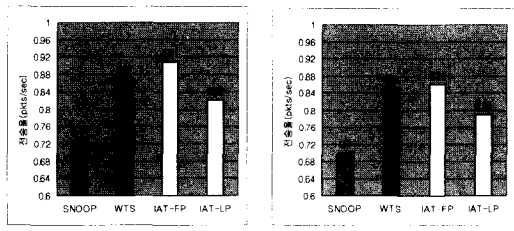
그림 8은 FTP에서 발생하는 데이타를 전송하는 시뮬레이션에서 시간이 진행됨에 따른 처리율(throughput)의 변화를 약 30초 동안 관찰한 것이다. 그래프의 가로축은 시간 축으로 단위는 초(second)이고, 세로축은 데이타 패킷의 시퀀스 번호이다. 먼저, 그림 8(a)는 기지국에 도착한 패킷의 시퀀스 번호를 보여주고 있는데, 이를 통해 각 스킴이 유선 링크의 자원을 얼마나 활용하였는지를 볼 수 있다. 송신단은 Indirect-ACK과 수신단으로부터 전송된 ACK 패킷을 구별하지 못하므로 제안하는 방안들에서는 기지국에서의 버퍼 임계치 조건이 허용하는 한은 데이타 전송을 계속 진행할 수 있다. 따라서 제안하는 방안들이 기존의 Snoop에 비하여 송신단 휴지 시간을 단축하고 유선 링크의 자원을 더 효율적으로 사용할 수 있음을 볼 수 있다. Snoop이 10.8-15.5 초 사이에서 긴 송신단 휴지시간을 보이는 동안 제안하는 세 방안은 지속적으로 데이타 패킷을 전송하는 것을 볼 수 있다. 그림 8(b)는 송신단과 수신단 사이에 성공적으로 전송된 데이타 패킷의 양을 보여주고 있는데, 이를 통해 종단간 연결의 처리율을 비교할 수 있다. Indirect-ACK 기법을 사용하는 세 방안들이 Snoop 송신단의 휴지 시간을 줄이고 유선 링크의 자원을 보다 효율적으로 사용하기 때문에 유·무선 통합 네트워크에서의 TCP 연결의 종단간 성능을 향상시킴을 볼 수 있다.

그림 9와 10은 각각 FTP와 TELNET 응용의 경우에 대하여 각 스킴의 패킷 전송률을 비교하는 그래프이다. 이들 결과는 15번의 실험을 통한 평균값이며, 각 실험은 전송이 시작된 직후부터 약 30초간 실시되었다. 그래프의 세로축은 1초 동안 전송된 패킷의 수로 단위는 packets/sec 이다. 그림 9(a)와 10(a)는 송신단과 기지국 사이의 유선 링크에서의 패킷 전송률을 보여준 것인데, 이를 통해 각 스킴의 유선 링크 활용 정도를 볼 수 있다. 반면에 그림 9(b)와 10(b)는 종단간 연결에서의 패킷 전송률을 보여주고 있는데, 이를 통해서는 각 스킴의 궁극적인 처리율을



(a) 유선 링크의 패킷 전송률 (b) 종단간 패킷 전송률

그림 9 FTP 응용에 대한 패킷 전송률 비교



(a) 유선 링크의 패킷 전송률 (b) 종단간 패킷 전송률

그림 10 TELNET 응용에 대한 패킷 전송률 비교

비교할 수 있다.

FTP와 TELNET 두 경우 모두 상대적으로 유선 링크의 활용률이 높은 스킴이 종단간 처리율도 높은 것을 볼 수 있다. 그런데, TELNET의 경우는 Snoop에 비한 제안하는 스킴들의 유선 링크의 패킷 전송률 향상 정도와 종단간 패킷 전송률의 향상 정도가 거의 비슷하지만 FTP의 경우는 종단간 패킷 전송률 향상 정도가 유선 링크의 패킷 전송률 향상 정도에 미치지 못하는 것을 볼 수 있다. 즉, FTP와 같이 연속적인 데이타가 있는 응용에 대해서는 제안하는 스킴들이 획득하는 유선 링크 활용률 향상이 종단간 처리율 향상으로 모두 이어지지 못하는 것을 알 수 있다. 또한, FTP의 경우는 WTS가 가장 큰 성능 향상을 가져오는데 반해 TELNET의 경우는 WTS와 IAT-FP의 성능 향상 정도가 거의 비슷하다. 즉, FTP와 같이 연속적으로 데이타 패킷을 발생시키는 응용에 대해서는 WTS와 같이 일찍(무선 링크에서의 패킷 손실을 발견하는 즉시) 그리고 오류가 회복될 때까지 계속 Indirect-ACK을 발송하는 것이 유리함을 알 수 있다. FTP와 TELNET 두 응용에 대하여 모두 제안하는 스킴들은 Snoop에 비해 종단간 패킷 전송률을 약 20% 정도까지 향상시킬 수 있었다.

그림 11과 그림 12는 TCP 송신단의 윈도우 크기를 4로 하였을 때, 30초의 실험 시간 동안 제안하는 각 스킴의 경우 기지국에서 평균적으로 사용하는 버퍼의 크기와 최대로 사용하는 버퍼의 크기를 비교한 것이다. Snoop 프로토콜을 적용한 기지국은 TCP의 최대 혼잡 윈도우 크기(Congestion Window Size)만큼의 버퍼를 필요로 하는 반면, 제안하는 세 스킴은 일시적으로 Indirect-ACK 패킷을 사용하여 송신단과 기지국, 기지국과 수신단 사이의 TCP 연결을 각각 별도로 제어하므로 Snoop 프로토콜을 적용한 경우보다 기지국 버퍼 오버헤드가 크다. 각 응용별로 필요로 하는 버퍼 양의 차이에 비하여 동일한 응용의 경우에 스킴 간 차이는 거의 없음을 볼 수 있다. FTP의 경우는 평균적으로 최대 혼잡 윈도우 크기에 해당하는 버퍼를 사용하고, 최대로는 최대 혼잡 윈도우 크기의 2배 정도에 해당하는 버퍼를 사용한다. 반면, TELNET의 경우에는 최대로 사용하는 버퍼 양이 최대 혼잡 윈도우 크기 정도에 해당한다.

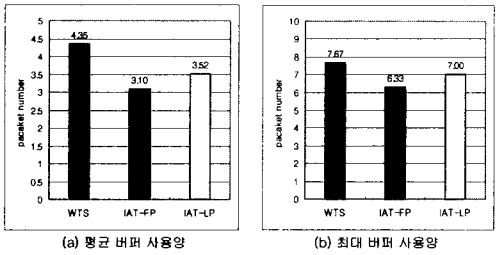


그림 11 FTP 응용의 경우 버퍼 사용양 비교

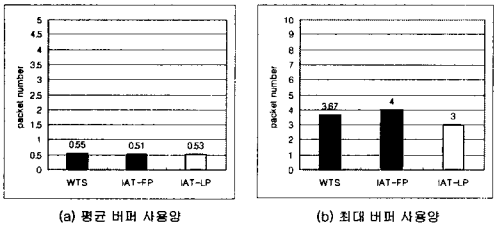


그림 12 TELNET 응용의 경우 버퍼 사용양 비교

그림 9, 10과 그림 11, 12를 종합적으로 볼 때, WTS는 IAT-FP와 같은 Indirect-ACK 타이머 사용 오버헤드 없이 IAT-FP와 유사한 양의 버퍼 오버헤드로 IAT-FP와 비슷하거나 IAT-FP보다 더 나은 종단간 패킷 전송률 향상을 가져올 수 있음을 볼 수 있다. 특히 FTP와 같이 응용 데이타가 연속적으로 발생하는 경우에는 IAT-FP와 거의 유사한 버퍼 오버헤드로 IAT-FP보다 훨씬 나은 성능을 얻을 수 있다.

5. 결 론

TCP는 전송 비트 오류에 의한 패킷 손실 확률이 매우 낮은 유선 망을 대상으로 설계된 프로토콜이므로, 이를 그대로 유·무선 통합 환경에 적용할 경우 TCP 송신단이 무선 망에서 발생한 전송 비트 오류로 인한 패킷 손실도 네트워크 혼잡에 의한 것으로 가정하여 송신단 전송률을 낮추기 때문에 성능이 저하하게 된다. 이에 유·무선 통합 망을 위한 TCP 전송률 향상 방안이 여러 가지로 제안되었는데, 본 연구에서는 기존에 제안된 방안들 가운데 Snoop에 비해 무선 네트워크에서 발생한 패킷 손실을 송신단에게 감추는 능력을 향상시키면서 일시적인 종단간 연결분리만을 허용하는 방안을 제안하였다. 제안하는 방안들은 기본적으로 Snoop과 같이 작동하지만 송신단에서 무선 네트워크에서의 패킷 손실을 발견할 염려가 있다고 판단되는 경우에는 제한적으로 Indirect-ACK을 사용하는데, 그 판단 기준에 따라 WTS, IAT-FP, IAT-LP등의 세 가지 스킴으로 구분된다. 제안하는 방안들은 기지국 버퍼에 임계치를 두어 기지국 버퍼의 길이가 이 임계치 이하일 때에만 Indirect-ACK을 사용하도록 제한함으로써 Indirect-ACK 사용으로 인한 버퍼 오버헤드를 제한한다.

시뮬레이션을 통해 제안한 스킴들을 비교한 결과 버퍼 오버헤드 측면에서는 제안하는 세 가지의 스킴이 모두 거

의 유사하지만, 구조의 복잡성 및 성능 향상 측면에서는 WTS가 가장 우수하였고, 특히 연속적인 패킷을 발생시키는 FTP와 같은 응용에 대해서는 WTS의 성능이 가장 좋았다. 또한, 제안한 스킴들과 Snoop의 성능을 비교한 결과 제안한 스킴들이 최대 Snoop의 2배 정도 되는 버퍼를 사용함으로써 Snoop에 비하여 TCP 성능을 10~20% 정도 향상시킬 수 있음을 확인할 수 있었다.

참 고 문 헌

[1] W. R. Stevens, *TCP/IP* Illustrated, Vol. 1, Addison-Wesley, Nov. 1994.

[2] A. Bakre and B. R. Badrinath, I-TCP: Indirect TCP for Mobile Hosts, Proceedings of the 15th International Conference on Distributed Computing Systems, pp. 136-143, June 1995.

[3] A. Bakre, B.R. *Badrinath, Handoff and System Support for Indirect TCP/IP*, Second Usenix Symposium on Mobile and Location-dependent computing, Ann Arbor, Michigan April 1995.

[4] Bakre, A.V. Badrinath, B.R, *Implementation and Performance Evaluation of Indirect TCP*, Computers, IEEE Transations on, Volume:46 Issue: 3, March 1997.

[5] H. Balakrishnan, S. Seshan, and R. H. Katz, *Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks*, ACM Wireless Networks, Dec. 1995.

[6] Hari Balakrishnan, Srinivasan Seshan, Elan Amir, Randy H. Katz. *Improving TCP/IP Performance over Wireless Networks*, Proc. 1st ACM Conf. on Mobile Computing and Networking, Berkeley, CA, November 1995.

[7] Balakrishnan, H.; Padmanabhan, V.N.; Seshan, S.; Katz, R.H. *A comparison of mechanisms for improving TCP performance over wireless links, Networking*, IEEE/ACM Transactions on , Volume: 5 Issue: 6 , Dec. 1997.

[8] Karu Ratnam and Ibrahim Matta, WTCP: *An Efficient Transmission Control Protocol for Networks with Wireless Links*, Third IEEE Symposium on Computer and Communications (ISCC '98).

[9] Karu Ratnam, Ibrahim Matta, *Effect of Local Retransmission at Wireless Access Points on the Round Trip Time Estimation of TCP*, Proceedings of the The 31st Annual Simulation Symposium '98

[10] K. Brown and S. Singh, *M-TCP: TCP for Mobile Cellular Networks*,, ACM CCR Vol. 27(5), 1997.

[11] Elaoud, M.; Ramanathan, P, TCP-SMART: *A technique for improving TCP performance in a spotty wideband environment*, Communications, 2000. ICC 2000.

[12] 김종덕, 임효준, 김종권, "무선 망 위에서의 TCP 성능 향상 방법과 성능 분석", 1999년도 한국통신학회 추계 종합학술 발표회.

[13] J. Border, M. Kojo, J. Grainer, G. Montenegro and Z. Shelby, *Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations*, IETF RFC 3135, June 2001.

[14] Network Simulator Document by U.C. Berkely http://www.isi.edu/nsnam/ns/

[15] 조준상, 최명환, "전송오류율이 높은 무선환경에서의 TCP 성능 저하 극복 방안", Telecommunications Review 제10권 6호, 11~12월, 2000.

[16] Jochen Schiller, *Mobile Communications*, Addison-Wesley, 2000.

김 윤 주

2000년 이화여자대학교 컴퓨터학과(학사) 2002년 이화여자대학교 대학원 컴퓨터학과(석사). 2002년~현재 한국전자통신연구원 무선방송연구소 무선LAN연구팀 관심분야는 무선 망 프로토콜, 무선 MAC 프로토콜, 무선 LAN, 이동통신 네트워크

이 미 정

1983~1987년 이화여자대학교 전자계산학 학사. 1987~1989년 University of North Carolina at Chapel Hill 컴퓨터학 석사. 1990~1994년 North Carolina State University 컴퓨터공학 박사. 1994~현재 이화여자대학교 공과대학 컴퓨터학과 부교수. 관심분야는 프로토콜 설계 및 성능 분석, 멀티미디어 전송을 위한 트래픽 제어, 인터넷에서의 QoS 지원, 트래픽 엔지니어링, 무선 이동 네트워크, Ad-hoc 네트워크

안 재 영

1983년 연세대학교 전기공학과(학사) 1985년 연세대학교 대학원 전기공학과(석사). 1989년 연세대학교 대학원 전기공학과(박사). 1989년~현재 한국전자통신연구원 무선방송연구소 무선LAN연구팀 팀장/책임연구원. 관심분야는 디지털 통신, 무선 MAC 프로토콜, 무선 LAN, 이동통신 네트워크