

이동 인터넷에서의 TCP와 무선데이터링크의 재전송 간섭

조 현 준[†]

요 약

TCP 프로토콜은 패킷 손실이 주로 혼잡에 의하여 발생하는 유선환경에 맞추어 설계되었기 때문에, 무선링크 상의 비교적 높은 에러 발생에 의하여 이동인터넷에서는 그 성능이 낮아질 수 있다. 본 논문에서는 TCP의 성능감소 원인인 재전송 간섭이 성능에 미치는 영향에 대하여 다룬다. 성능분석을 위하여 무선망을 포함하는 인터넷 환경에 대하여 시뮬레이션 모델을 만들고 PARSEC을 이용하여 시뮬레이션 프로그램을 작성한 후 이를 수행하였다. 또한, 시뮬레이션결과를 바탕으로 재전송 간섭현상을 줄일 수 있는 방안을 제시하였다.

Retransmission Interference Between TCP and Wireless Data Link in Mobile Internet

Hyun-Joon Cho[†]

ABSTRACT

TCP protocol is originally designed for the wired network in which the packet loss is mostly caused the network congestion, and thus the performance of protocol can be decreased in mobile Internet environment due to the comparatively high error rate of wireless link. This paper analyzes the influence of retransmission interference that is a major factor to the TCP performance decrease. We first build a model in order to simulate the mobile Internet environment that includes the wireless network, and then we implement a program using PARSEC simulation language and finally executed the program. Based on the simulation results, we suggest a method that reduce the effect of the retransmission interference.

1. 서 론

최근 들어 무선통신 기술의 발달에 의하여 유선 망으로 구성된 현재의 인터넷은 가까운 미래에 무선 망을 포함한 이동 인터넷(mobile Internet) 형태로 변하게 될 것으로 예상됨에 따라, 전송프로토콜인 TCP 개선에 대한 많은 연구들이 이루어지고 있다[1-3]. 즉, 오늘날의 인터넷은 유선 망에 기반을 두고 있기 때문

에 무선망이 인터넷에 포함될 경우에 기존 유선망 프로토콜이 적절하게 동작할 수 있는가에 대한 연구와 프로토콜 수정에 대한 연구가 필요하다. 특히 인터넷에서 사용되고 있는 전송프로토콜(transport protocol)인 TCP(Transmission Control Protocol)은 유선 망과 고정호스트로 이루어진 망에 적합하도록 설계되었기 때문에 이를 그대로 무선망에 적용하면 불필요한 혼잡 제어 및 회피 동작 등으로 인한 중단대중단(end-to-end) 처리율의 성능저하를 초래한다. 그 이유는 TCP가 전송도중에 발생하는 패킷에 대한 손실을 혼잡

[†] 종신회원 : 전주대학교 전자매체공학부 교수
논문접수 : 1999년 10월 15일, 심사완료 : 1999년 11월 12일

(congestion)에 의한 것으로 간주하기 때문이다. 일단 혼잡이 발생한 것으로 판단하면 전송 윈도우의 크기를 줄이고 손실된 패킷을 재전송하며, 재전송을 위한 타임어의 값을 증가시킨다. 그러나 무선망에서의 패킷 손실은 무선망 자체의 높은 비트 에러율에 의하여 발생한다.

무선링크는 일반적으로 유선링크에 비하여 에러율이 높다. 무선링크의 신뢰성을 높이기 위한 자연적인 방법은 데이터링크 계층에서의 재전송을 이용하는 것이다. 데이터링크 계층에서의 재전송은 일반적으로 이동호스트(MH: Mobile Host)가 유선 망에 연결되어 있는 기지국(BS: Base Station)으로부터 전송된 각 메시지에 대하여 응답메시지(ack)를 전송함으로써 이루어진다.

만일 기지국이 타임아웃 기간 동안에 기대되는 응답 메시지를 받지 못하면 해당 메시지를 재전송한다. 이러한 과정을 응답메시지를 받을 때까지 계속 반복한다.

데이터링크 계층에서의 재전송은 상위계층에게 보다 높은 신뢰성을 제공한다. 그러나 링크계층의 재전송은 TCP 계층에서의 재전송과 간섭을 일으켜서 TCP 성능에 나쁜 영향을 미친다[3]. 이에 따라 TCP 재전송과 간섭을 일으키지 않는 기지국으로부터 재전송으로 구성된 국부회복(local recovery)에 대하여 많은 연구가 수행되었다. 그러나 이들의 연구결과는 기지국의 프로토콜 기능을 상당부분 수정해야만 가능한 것들이 대부분이어서 구현상에 어려움이 많은 것들이다[4].

본 논문에서는 무선 재전송 간섭이 TCP 성능에 미치는 영향에 대하여 시뮬레이션을 통하여 분석해보고 이를 토대로 간섭효과를 줄일 수 있는 비교적 단순한 방안을 제안한다. 논문의 구성은 다음과 같은 순서로 이루어져 있다. 1장 서론에 이어서 2장에서는 지금까지 연구된 무선망 TCP의 성능향상을 위한 연구들을 살펴본다. 3장에서는 무선링크를 갖는 인터넷에서의 TCP 성능분석을 위한 시뮬레이션 방안에 대하여 기술하고 그 결과를 분석한다. 그리고 성능을 향상시킬 수 있는 방안을 제안한다. 마지막으로 4장에서 결론을 맺는다.

2. 관련연구

2.1 무선링크를 갖는 TCP 성능향상에 대한 기존 연구 무선링크를 갖는 망에서의 TCP 성능개선을 위한 많

은 연구들이 있었다. Indirect TCP는 TCP 연결을 유선 망에서의 경로와 무선 링크의 두 부분으로 나누어서 유선 망에서는 기존 TCP를 수행하고 무선링크 상에서는 무선특성을 고려한 TCP를 사용하는 방법이다[4]. 이 방안의 단점은 TCP의 종단-대-종단 시맨틱(end-to-end semantic)이 유지되지 못한다는 점이다[1-3].

Snoop 프로토콜은 이동호스트를 오가는 모든 패킷들을 캐싱했다가 전송 손실이 감지되면 이전에 캐싱한 패킷을 재전송함으로써 재전송효율을 높이는 방안이다. 또한 무선링크에서 에러가 발생하여 재전송을 수행하는 동안 수신 단으로부터 전송되어 온 중복 응답을 기지국이 가로채서 불필요한 혼잡제어를 막는 방안이다[5]. 이 방안은 종단대종단 시맨틱을 유지하면서 indirect TCP보다 더 좋은 성능을 보인다.

Explicit Bad State Notification(EBSN) 방법은 무선 채널 상에서 버스티(bursty) 에러 상태가 검출되면 기지국은 송신 단에 특정메시지를 보내어 불필요한 혼잡제어에 따른 전송률 저하를 막는 방안이다[2].

이상에서와 같은 방안들은 송신 단과 수신 단 사이에 있는 기지국과 많은 상호작용을 요구하며, 기존의 프로토콜 구조에 상당부분 수정을 요구한다. 따라서 현실적으로 적용에 어려움이 많다. 또한, IP 패킷이 암호화되었을 경우에는 TCP 패킷과 다른 사용자 패킷을 기지국이 구분하기 힘들고, TCP 패킷에 관련된 순서번호 등을 사용할 수 없다. 데이터패킷과 응답 패킷이 전송되는 경로가 다를 경우, 특히 위성통신의 경우에서 동일한 중간노드(기지국)를 요구하는 것은 문제가 된다. 따라서 본 논문에서는 가급적 기지국에서 수행할 임무를 줄이고 송, 수신 단의 TCP 기능을 간단히 수정하여 재전송의 간섭현상을 피할 수 있는 방안에 대하여 다룬다.

2.2 TCP의 재전송[6]

수신 측의 TCP는 송신 측으로부터 전송되어 온 각 데이터 패킷을 받을 때마다 응답을 송신 측으로 전송함으로써 종단대종단 시맨틱을 유지한다. 만일 송신 측이 전송한 패킷이 수신 측에 도착하기 전에 손실되면 송신 측은 이를 재전송 해야 한다. 이 재전송을 위해서 TCP 송신 측에서는 round trip time(RTT)를 계속적으로 측정하여 재전송을 위한 타임아웃(timeout) 값을 다음 식에 의하여 계속적으로 계산한다.

$$RTT = (\alpha * Old_RTT) + ((1 - \alpha) * New_Round_Trip_Sample) \quad (0 < \alpha < 1)$$

$$Timeout = \beta * RTT \quad (1 < \beta \leq 2)$$

α 는 새로 측정된 RTT의 값(New_Round_Trip_Sample)과 이전에 계산된 RTT(Old_RTT)의 값을 고려하여 새로 계산된 RTT 값을 결정할 때 사용되는 가중치이며, β 는 재전송타임아웃값과 계산된 RTT 값의 비를 의미한다.

이때 RRT의 측정값이 정확해야 하기 때문에 한 패킷에 대하여 연속적으로 전송이 실패했을 경우에는 RTT를 측정하지 않으며, 타임어 백오프(backoff) 방법으로 결정하게 된다. 그 이유는 연속적인 실패는 혼잡 현상에 의하여 발생하므로 다음 전송에 필요한 재전송 타임아웃의 값을 증가시켜야만 연속적인 실패를 피할 수 있기 때문이다. 일반적으로 사용되는 백오프 방법은 다음 식과 같다.

$$new_timeout = \gamma * timeout \quad (\gamma = 2)$$

즉, 새로운 재전송 타임아웃값(new_timeout)은 기존 재전송 타임아웃값의 2배로 계산된다. 이상에서와 같은 적응적인 방법으로 재전송을 위한 타임아웃 기법을 사용함으로써 망에서 혼잡이 발생했을 때 이를 피할 수 있게 된다. 이와 함께 망 대역폭을 효과적으로 사용하기 위한 흐름제어 기법들이 연구, 발전되어 현재 많이 사용하고 있는 Tahoe, Reno 등의 TCP 버전이 나타났다[8, 9].

2.2.1 Tahoe TCP

Tahoe 버전 TCP는 흐름제어 기법으로 슬로우 스타트(slow start), 혼잡회피(congestion avoidance), 신속한 재전송(fast retransmit) 기법을 사용한다. 각 TCP 연결마다 혼잡 윈도우(congestion window)와 슬로우 스타트 경계를 나타내는 슬로우 스타트 경계값(slow start threshold)를 이용하여 송신 측에서 흐름제어를 수행한다.

연결 초기에는 혼잡 윈도우의 크기를 1 세그먼트로 하여 전송을 시작한 후 혼잡 현상이 일어나기 전까지는 슬로우 스타트 기법에 따라 전송 흐름을 제어한다. 만약 혼잡 현상이 발생하여 재전송 타임아웃이 발생하거나 중복응답이 수신되면 혼잡현상을 감지하여, 슬로우 스타트 경계값에 혼잡 윈도우의 크기의 절반을 대입한다. 그리고 혼잡 윈도우의 크기를 1 세그먼트로 하

여 슬로우 스타트 기법을 수행한다. 만약 윈도우의 크기가 슬로우 스타트 경계 값에 도달하면 혼잡회피 알고리즘이 수행된다. 슬로우 스타트 기법에서는 하나의 응답을 받을 때마다 윈도우의 크기를 한 세그먼트씩 증가시키며, 혼잡회피 알고리즘에서는 왕복지연시간 동안에 1 세그먼트 크기만큼만 정제 윈도우를 증가시키는 방법이다.

신속한 재전송이란 동일한 세그먼트에 대한 응답을 3번 수신하게 되면 재전송 타임아웃을 기다리지 않고 해당 세그먼트가 손실되었다고 판단하여 재전송 하는 방법이다. 재전송이 시작되면 앞에서 언급한 슬로우 스타트와 혼잡회피 기법이 사용된다.

2.2.2 Reno TCP

세 개의 중복된 응답을 받았을 때 Tahoe 버전에서와 같은 슬로우 스타트 기법을 수행하는 것이 아니고 정제 회피 기법을 이용한다. 즉, 3개의 중복된 응답을 송신 측에서 받았을 때, 슬로우 스타트 경계 값을 현재의 혼잡 윈도우의 절반 값으로 설정한다. 만약 그 값이 2보다 작으면 2로 설정한다. 또한 혼잡 윈도우를 슬로우 스타트 경계 값에 3을 더한 값으로 설정한다. 그 후 손실된 세그먼트를 전송하고 중복된 응답이 올 때마다 혼잡 윈도우를 1 세그먼트씩 증가시키면서 새로운 세그먼트를 전송한다. 새로 설정된 윈도우에 포함되는 세그먼트가 수신되면 혼잡 윈도우를 슬로우 스타트 경계 값으로 설정하고 정제 회피 기법을 수행한다. 이렇게 하므로써 슬로우 스타트 기법에서 대역폭이 낭비되는 현상을 피할 수 있다. 일반적으로 한 세그먼트의 손실에 따른 재전송 기법으로는 Reno 버전이 Tahoe 버전보다 더 좋은 성능을 나타내는 것으로 알려져 있다.

이외에도 Reno 버전에 선택적 재전송 기법(selective repeat retransmission)을 도입한 Reno with SACK 버전이 있다. 이 방안은 손실이 발생한 세그먼트만을 재전송하기 때문에 한 윈도우 내에 존재하는 다수의 세그먼트들이 손실되었을 경우에 보다 좋은 성능을 나타낸다.

2.3 TCP와 무선 데이터 링크 재전송의 간섭현상

TCP프로토콜은 송신 측으로부터의 재전송을 유발하는 두 가지 경우가 있다. 첫째, 재전송 타임아웃 기간 내에 특정 패킷에 대한 응답이 도착하지 않았을 경우

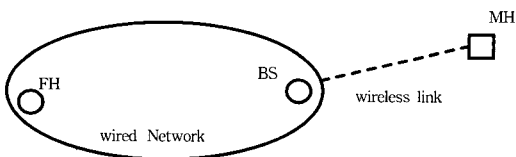
송신 측에서는 패킷이 손실된 것으로 가정하고 재전송을 수행한다. 두 번째, 신속한 재전송 기능에 따라 한 패킷에 대한 3개의 중복응답을 받으면 송신 측은 패킷이 손실된 것으로 판단하고 패킷을 재전송 한다. 두 경우 모두에 대해서 기존 TCP 프로토콜은 망에 혼잡이 발생한 것으로 간주하고, 혼잡제어 기능이 동작한다. 즉, 윈도우 크기를 줄이고 전송 성능을 떨어뜨리게 된다. 유선 망에서는 전송에러의 대부분이 망 혼잡에 따른 현상이기 때문이다. 그러나 무선링크에서의 패킷 유실일 경우 이와 같은 현상은 바람직하지 않다. 무선링크에서의 패킷유실은 일반적으로 데이터링크 계층에서 재전송에 의하여 처리하는 것이 바람직하다. 그러나 데이터링크 계층에서 재전송 하는 도중에 TCP 프로토콜에서 재전송 타임아웃이나 3개의 중복응답에 따른 재전송이 수행될 수 있는데 이를 간섭(interference) 현상이라 한다[3].

3. 시뮬레이션

3.1 시뮬레이션 가정

시뮬레이션은 UCLA의 parallel computing Lab.에서 만든 범용 시뮬레이션 도구인 PARSEC[7]을 이용하여 수행하였으며, 기본가정은 다음과 같다.

첫째, 시뮬레이션의 대상이 되는 이동인터넷의 망의 형태는 기존 연구에서 가정하는 것과 동일한 그림 1과 같다고 가정한다. (그림 1)에서 송신단은 고정호스트(FH)이며 수신단은 이동호스트(MH)이다.



(그림 1) 유선 망과 무선 링크로 이루어진 이동인터넷망

둘째, TCP 프로토콜은 Reno 버전을 따른다.

세째, 기지국(BS) 상에서의 데이터링크 기능이 go-back-N 재전송 메카니즘을 수행한다고 가정한다. 그러므로 기지국에서 이동 호스트까지의 전송은 순서가 보장된다.

넷째, 이동 호스트(MH)의 TCP 프로토콜은 매번 패

킷을 수신할 때 마다 응답을 전송한다고 가정한다.

다섯째, 전송되는 패킷들의 순서는 보장된다고 가정한다.

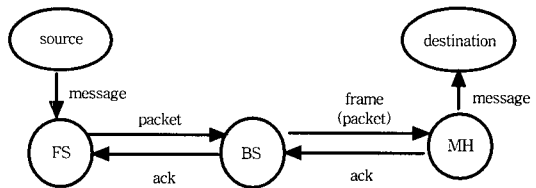
여섯째, 유선 망이나 무선링크에서의 지연은 일정하다고 가정한다.

일곱째, 전송되는 패킷은 항상 일정한 크기를 가지며, 패킷 생성 시간 간격은 지수분포에 따른다.

여덟째, 기지국이나 이동호스트의 수신버퍼의 크기는 64Kbyte라고 가정한다. 버퍼에서 오버플로우가 발생할 경우 뒤에 수신된 패킷들은 버려진다.

3.2 시뮬레이션 모델

본 논문에서 수행하는 시뮬레이션의 모델은 (그림 1)의 망 요소를 고려한 다음 (그림 2)와 같다.



(그림 2) 시뮬레이션 모델

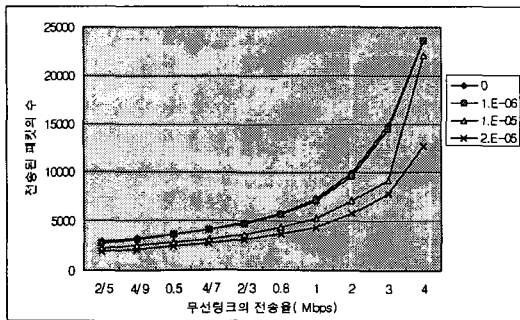
(그림 2)에서 소스(source)는 패킷을 생성하여 FS에게 전송을 요구하는 부분으로 TCP 프로토콜의 응용 프로세스 역할을 한다. FS는 송신 단의 TCP 프로토콜 모듈의 기능을 수행하며, BS에게 패킷을 전송한다. 이때 전송 율과 전파지연시간은 FS와 BS가 포함되어 있는 유선 망의 성능에 의하여 결정된다. 또한 BS와 MS 사이의 전송은 무선링크 상에서 전송되므로 무선링크의 전송 율과 전파지연, 그리고 무선링크의 에러발생 확률 등의 파라미터에 의하여 성능이 결정된다. MH와 목적지(destination) 사이에서는 전송된 메시지가 전달되며, MH는 수신 단의 TCP 프로토콜 모듈의 기능이 주로 처리된다. 목적지는 수신 단의 TCP 프로토콜의 응용 프로세스 역할을 하게 된다.

무선링크는 go-back-N ARQ 기법을 이용하는 것으로 가정하였기 때문에 전송도중에 링크 상에서 비트 에러가 발생하면 패킷전송이 실패하게되고 에러검출이나 재전송 타임아웃에 의하여 실패된 패킷 이후부터 다시 전송이 이루어진다. 따라서 에러가 없을 경우 패킷들은 링크 전송 율에 따라 순차적으로 전송되고 링

크전송지연만큼 지연이 발생한 후 이동호스트에 도착하게 된다. 에러가 발생한 경우는 재전송에 따른 지연만큼 더 지연이 발생하는 것을 제외하고는 에러가 없는 경우와 동일하게 시뮬레이션 될 수 있다.

3.3 시뮬레이션 및 결과분석

전송되는 패킷의 크기는 1Kbyte로, 응답의 크기는 40byte로 일정하다고 가정하며, 유선 망에서의 전송율은 4Mbps, 유선망에서의 지연시간은 50msec 라고 가정한다. 본 연구의 목적은 TCP와 무선링크 상에서의 데이터링크 프로토콜 간의 재전송에 따른 간섭현상에 대한 성능분석이므로 유선 망에서의 특성 변화는 고려하지 않는다. 무선링크의 전송 율에 따른 throughput을 살펴보기 위하여 무선링크에서의 지연은 1msec라고 가정하고 시뮬레이션을 수행하였으며, 시뮬레이션 기간은 60초로 설정하였다. 또 무선링크의 비트 에러율에 따라 에러율이 각각 0, 10^{-6} , 10^{-5} , 2×10^{-5} 일 경우에 따라 실험을 수행하였다. 그 결과는 다음 (그림 3)과 같다.

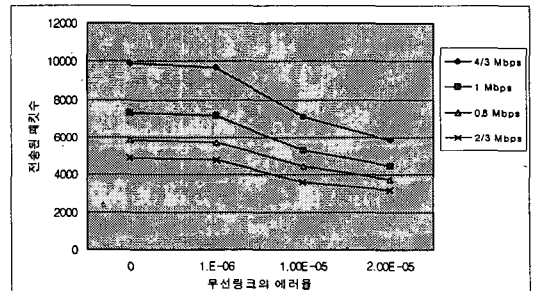


(그림 3) 무선링크의 전송 율에 따른 전송된 패킷의 수

(그림 3)에서 무선링크의 전송 율에 따라 전송된 패킷의 수의 변화가 나타나고 있으나, 이는 무선링크의 전송 율이 유선링크의 전송 율에 비해 낮기 때문에 패킷전송에 대한 전체적인 성능은 결국 무선링크 전송 율에 의하여 결정되기 때문이다. 즉, 무선링크의 전송 율에 비례해서 정해진 기간동안 전송된 패킷의 수는 증가비례할 수 밖에 없다. 이 그림에서 보면 같은 전송 율이라도 에러 율이 증가함에 따라 전송된 패킷의 수가 감소하는 것을 볼 수 있다. 이는 무선링크의 에

러 발생에 따른 무선데이터링크의 재전송과 이 때문에 생긴 지연으로 송신 단의 재전송 타임아웃의 발생으로 기인한 성능저하로 볼 수 있다. 이 현상이 본 논문에서 관심을 갖는 재전송 간섭현상이다.

무선링크의 비트 에러 율이 10^{-6} 보다 낮을 경우에는 에러 율이 0인 경우와 거의 비슷한 성능을 보이는데, 이는 아주 간헐적인 무선링크의 에러발생에 따른 재전송은 TCP 재전송을 야기할 만큼 심각하지 않다는 의미이다. 이는 TCP Reno 버전이 망에서 발생하는 지연에 적응성이 높기 때문이다. 그러나 에러율이 10^{-5} 이 되면 상당한 성능감소를 나타낼을 볼 수 있다. 에러 율에 따른 성능감소를 알아보기 위해 같은 4가지 무선링크의 전송 율에 대해서 에러 율에 따라 60초간 전송된 패킷의 수를 그래프로 나타낸 것이 다음 (그림 4)이다.

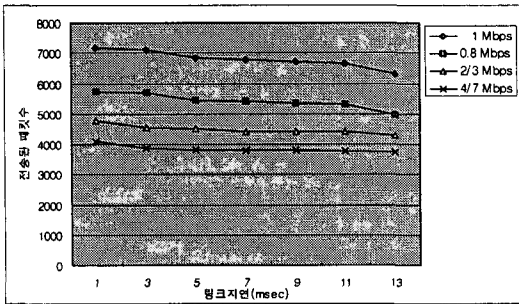


(그림 4) 무선링크의 에러 율에 변화에 따른 전송된 패킷 수

(그림 4)에서 선명히 나타나듯이 에러율의 증가는 성능에 상당한 영향을 주는 요소임을 알 수 있다.

한편, 무선링크의 지연 변화에 따른 성능측정을 하기 위하여 무선링크의 에러 율은 10^{-6} 으로 고정시키고 전송 율을 1 Mbps, 4/5 Mbps, 2/3 Mbps, 4/7 Mbps로 변화시키면서 전송된 패킷의 수를 측정하였다. 그 결과가 다음 (그림 5)이다.

(그림 5)에서 알 수 있듯이 무선링크 지연의 변화는 성능이 그리 큰 영향을 주지 않음을 알 수 있다. 물론 지연이 11msec 이상에서는 약간의 성능저하가 있지만, 일반적으로 중간노드 없이 이루어지는 무선링크에서의 지연은 아주 작다는 점을 감안할 때 10msec 이상의 지연은 위성링크와 같은 특별한 경우를 제외하고는 현실적이지 못하다.



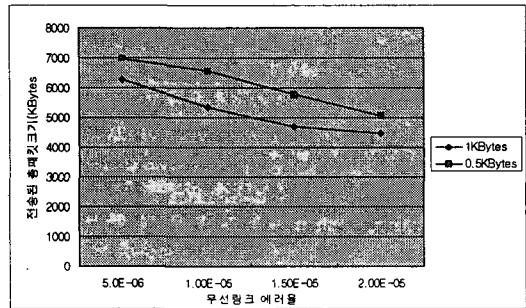
(그림 5) 무선링크지연에 따른 전송된 패킷 수

3.4 재전송 간섭을 줄이는 방안

앞에서 본 시뮬레이션 결과에서와 같이 재전송 간섭은 무선링크의 패킷 전송 시간보다는 비트 에러 율에 민감함을 알 수 있다. 또 무선링크의 지연시간은 여러 노드를 거쳐서 전송되는 유선 망에서의 총 지연시간에 비하여 일반적으로 매우 짧기 때문에 송신 단의 타임아웃에 따른 재전송에 크게 영향을 미칠 수 없으므로 고려의 대상이 아니다. 물론 위성링크와 같은 특별한 경우엔 다르지만 일반적인 무선링크의 경우엔 대부분 이러한 현상을 보인다. 그러므로 무선링크의 에러 율이 10^{-6} 이상으로 높을 경우 성능저하를 피하기 위한 방법으로는 무선 데이터 링크의 재전송이 진행중일 때에는 송신 단의 재전송타임아웃의 값을 증가시켜서 무선 데이터 링크 재전송에 따른 시간지연에 의하여 타임아웃이 발생하지 않도록 하는 방안이 있을 수 있다. 그러나 이 방안은 기존 EBSN 기법과 유사하며, 무선 데이터 링크와 TCP 계층간에 밀접한 상호작용을 요구하는 문제점이 있다. 보다 단순하며 현실적인 방법은 무선링크에서 패킷 전송의 에러 율을 저하시키기 위한 패킷크기를 줄이는 방안이다. 이 방법은 발생한 재전송 간섭의 효과를 줄인다고 보다는 재전송 간섭 발생을 줄이는 방안이며, 단순히 TCP 계층에서의 TCP 연결시에 수신측(이동호스트 측)에서 자신이 사용하는 무선링크의 특성에 따라 패킷 크기를 조절함으로써 얻을 수 있다. 다음 그림은 패킷의 크기를 1Kbytes에서 0.5Kbytes으로 줄였을 경우 동일한 시뮬레이션 환경에서 얻을 수 있는 단위시간당 전송된 총 바이트(byte) 수를 나타내었다.

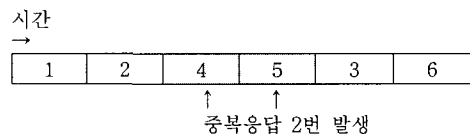
(그림 6)에서 알 수 있듯이 패킷의 크기를 절반으로 줄임에 따라 성능이 10% 이상 개선됨을 알 수 있다. 이는 패킷크기가 작아짐으로 생기는 시스템의 과부하

를 고려하더라도 높은 에러율 상황에서는 의미있는 성능개선이라 생각된다.



(그림 6) 패킷의 크기가 다른 두 경우에서의 성능비교

재전송 간섭의 두 번째 경우는 무선 데이터 링크에서 재전송을 수행하는 도중에 이동 호스트 쪽에서 고정호스트에게 중복응답을 3번 전송하는 경우이다. 이때 3번의 의미는 TCP 패킷을 전송하는 IP 경로가 다양할 수 있기 때문에 전송되는 패킷의 순서가 수신측에서 달라졌을 경우 2번까지 기다림으로써 수신측에서 재순서화(reordering)를 할 수 있도록 하기 위함이다. 다음 그림에서처럼 1번 패킷을 기다리는 상황에서 2, 3번이 도착한 후에 1번 패킷이 수신되면 2, 3번에 대한 응답은 중복응답이 된다. 중복응답을 송신측에서 받았을 때 곧바로 혼잡발생으로 간주하게되면 사실은 혼잡이 발생하지 않았기 때문에 대역폭을 효과적으로 사용하지 못하는 결과를 초래한다.



(그림 7) 중복응답 발생의 예

이를 제어하기 위하여 snoop 기법에서는 기지국에서 전송되는 중복응답을 가로채어 필요이상의 응답을 막는 방안을 제시하였다. 그러나 이는 기지국에서 IP 프로토콜이 사용자 데이터 부분에 있는 TCP 순서번호 등을 살펴서 처리해야 하는 문제점이 있다. 그러므로 현재의 프로토콜 구조에 상당한 수정을 요구하는 비현실적인 방법이다. 또 이동 호스트 측에서 중복응답을 지연하는 방안이 제시되기도 하였지만, 이동 호스트가

무선 데이터 링크의 재전송 유무를 알 수 없기 때문에 지연정도를 결정하기는 어렵다. 본 논문에서는 단순히 이동 호스트에서는 무선링크를 통해 정상적으로 수신된 패킷에 대해서만 응답을 보내는 방안을 제시한다. 무선링크의 재전송 메카니즘이 이동호스트에서 순서화된 신뢰성 있는 패킷 수신을 보장하므로, 수신순서에 대해서 이동호스트의 TCP 프로토콜은 부가적으로 처리할 것은 없다. 다만 무선링크 상에서의 재전송에 따른 지연된 패킷에 대한 응답을 정상적으로 수신 측에 도착할 때까지 보내지 않음으로써 불필요한 3개의 중복응답은 피할 수 있고 따라서 이 경우에 대한 간섭현상은 피할 수 있다.

4. 결 론

본 논문에서는 무선망을 포함하는 이동인터넷에서 기존 TCP 프로토콜의 성능을 저하시킬 수 있는 재전송 간섭현상의 영향에 대하여 분석하였다. 재전송 간섭현상은 무선 데이터 링크에서의 재전송과 TCP의 재전송이 중첩됨으로써 발생한다. 정량적인 분석을 위하여 이동인터넷 망에 대한 시뮬레이션 모델을 정하고, 범용 시뮬레이션 도구인 PARSEC을 이용하여 시뮬레이션 프로그램을 작성하였다. 또 시뮬레이션을 수행한 결과를 토대로 재전송간섭 현상을 줄이기 위한 방안으로 무선링크에서의 재전송 발생율을 줄이는 방법을 제시하였다.

재전송 간섭을 일으키는 요소로는 무선링크의 전송률, 지연, 에러율 등이 있는데 일반적으로 무선링크의 지연은 다수의 노드를 거치는 유선 망에 비해 무시할 수 있을 정도로 작기 때문에 나머지 전송율과 에러율을 파라미터로 하여 성능분석을 수행하였다. 시뮬레이션 결과 TCP의 흐름제어의 적응성 때문에 전송율의 변화는 재전송을 일으키는 요인이 되지 못하였다. 에러율의 경우는 비트 에러율이 10^{-6} 이하의 경우에는 무선링크의 재전송이 TCP 재전송을 유발시키지 못하였으며, 그 이상의 에러율 변화에는 비교적 민감한 반응을 보였다. 이와 같은 높은 에러율 상황에서 무선재전송의 발생을 줄일 수 있는 방안으로 TCP 패킷의 크기를 감소를 통한 패킷의 손실율을 낮추는 방안을 제시하였다. 시뮬레이션 결과 10% 이상의 성능향상을 확인할 수 있었다. 그러므로 이동 스테이션 측에서 TCP 연결설정 시에 이동링크의 에러율을 고려한 적

절한 TCP 패킷 크기를 결정하는 것이 기존 TCP 프로토콜의 수정없이 재전송 간섭을 피하는 간단한 해결책이 될 수 있다.

재전송 간섭의 두 번째 경우는 무선 데이터 링크에서 재전송을 수행하는 도중에 이동 호스트 측에서 고정호스트에게 중복응답을 3번 전송하는 경우이다. 본 논문에서는 단순히 이동 호스트에서는 무선링크를 통해 정상적으로 수신된 패킷에 대해서만 응답을 보내는 방안을 제시한다. 무선링크의 재전송 메카니즘이 기지국에서 이동호스트까지 순서화된 신뢰성 있는 패킷 전송을 보장하므로 수신순서에 대해서 이동호스트가 처리할 것은 없다. 다만 무선링크 상에서의 재전송에 따른 지연된 패킷에 대한 응답을 수신 측에서 정상적으로 수신될 때까지 지연시킴으로써 불필요한 3개의 중복응답은 피할 수 있다.

향후 이동인터넷의 성능에 많은 영향을 미치게 될 무선링크에서 사용될 효율적인 데이터링크제어 프로토콜과 위성링크와 같이 지연이나 에러율이 매우 큰 무선링크를 포함하는 인터넷에 대한 TCP 성능분석과 개선에 대한 연구가 계속적으로 필요하다.

참 고 문 헌

- [1] H. Balakrishnan, V. Padmanabhan, S. Seshan, and R. Katz, "A comparison of mechanism for improving TCP performance over wireless links," in ACM SIGCOMM, Stanford, CA, Aug. 1996.
- [2] B. Bakshi, P. Krishna, N. Baidya, and D. Pradhan, "Improving Performance of TCP over wireless networks," in 17th Int. Conf. Distributed Computing Systems ICDCS, May 1997.
- [3] Miten N. Mehta, Nitin H. Vaidya, "Delayed Duplicate-Acknowledgements : A proposal to Improve Performance of TCP on Wireless Links," Technical Report 98-006, Department of Computer Science, Texas A&M University,
- [4] A. Bakre and Badrinath, "I-TCP : indirect TCP for mobile hosts," in Proc. 15th International Conf. on Distributed Computing System(ICDCS), May 1995.
- [5] H. Balakrishnam, S. Seshan and R. H. Katz, "Improving Reliable Transport and Handoff Performance in Cellular Wireless Networks,"

Wireless Networks I, pp.469-481, 1995.

- [6] Douglas E. Comer, Internetworking with TCP/IP Volume 1, Prentice-Hall, 1995.
- [7] Richard A. Meyer, PARSEC User Manual(<http://pcl.cs.ucla.edu>), August 1998.
- [8] W. Stevens, "TCP Slow start, Congestion Avoidance. Fast Retransmit and Fast Recovery Algorithms," RFC 2001, 1997.
- [9] Kevin Fall and Sally Floyd, "Simulation-based Comparisons of Tahoe, Reno and SACK TCP," Computer Communication Review, Vol.26, No.3, pp.5-12, 1996.



조 현 준

e-mail : chohj@jeonju.ac.kr

1985년 고려대학교 전자공학과
졸업(학사)

1987년 고려대학교 대학원 전자
공학과 (석사)

1995년 고려대학교 대학원 전자
공학과 (박사)

1989년~1991년 (주)쌍용컴퓨터 시스템연구소 연구원
1995년~현재 전주대학교 전자매체공학부 조교수
관심분야 : 컴퓨터네트워크, 지능망프로토콜, 인터넷프
로토콜 등