

무선 TCP 성능증대를 위한 향상된 대역폭 측정 방법[☆]

Enhanced Bandwidth Estimation for TCP Westwood

강 문 수* 신 석 주**
Moonsoo Kang Seokjoo Shin

요 약

무선망에서 TCP의 성능을 증대시키기 위한 많은 연구가 진행되어 왔지만 패킷 손실률이 큰 경우, TCP의 성능은 아직도 개선의 여지가 많다. 본 논문은 무선 TCP의 한 종류인 TCP Westwood에서 랜덤 패킷 손실률이 높을 때 성능이 저하되는 문제를 지적하고 그 원인을 분석하였다. TCP Westwood의 성능 저하는 랜덤 패킷 손실률이 증가할수록, TCP Westwood의 가용 대역폭 측정(Available Bandwidth Estimation)의 정확도가 떨어지기 때문이다. 분석된 원인을 바탕으로 2가지의 향상된 가용대역폭 측정방법을 제안하고 그 성능을 시뮬레이션을 통해 확인하였다. 제안된 방법들은 무선 네트워크가 겪을 수 있는 랜덤 패킷 손실률 1% 이내에서 기존방법에 비해 대역폭 측정의 정확도가 향상되었다.

Abstract

Even though a lot of researches to improve the performance of TCP over wireless networks have been performed, the performance of TCP cannot satisfy users under high packet loss rate of wireless networks. This paper addresses the problem of TCP Westwood under high packet loss rate and reveals the reasons through the extensive simulation analysis. The ABE(Available Bandwidth Estimation) of TCP westwood becomes incorrect due to TCP dynamics of loss recovery. Based on this observation, we propose two algorithms for enhancing the correctness of ABE. We have evaluated the effects of the proposed algorithms through simulation. The proposed algorithms have achieved more correct ABE under the packet loss rate less than 1%.

☞ keyword : TCP, 무선 네트워크, 대역폭 측정

1. 서 론

인터넷의 폭발적인 성공은 TCP/IP로 불리는 인터넷 프로토콜 구조의 안정성에 기인한다. 그 중에서도 TCP (Transmission Control Protocol)은 인터넷에서 종대종 간의 신뢰적 데이터 전송 및 네트워크 혼잡제어 기능을 성공적으로 담당하고 있다. 그러나 새롭게 도입되고 있는 무선 네트워크에서 TCP는 기존 유선망에서 보여주었던 안정적인 성능을 보여주지 못하고 있다.

무선망에서 이러한 TCP의 성능저하는 무선망의 높은 손실률에 기인한다. 유선망에서는 전송 매체의 손실률(BER: Bit Error Rate)이 10^{-7} 정도로 미미하여, 전송 매체의 문제로 인한 패킷 손실은 거의 없다. 따라서 패킷 손실은 대부분 혼잡(Congestion)에 따른 라우터의 버퍼 오버플로우 때문에 발생한다. 이와 같은 유선망의 특징을 고려하여, 현재의 TCP(Tahoe, Reno, NewReno, SACK)는 패킷 손실을 곧 네트워크 혼잡으로 판단하며, 패킷 손실 발견 시, 손실 패킷의 재전송 및 네트워크 혼잡제어를 시도한다.

이러한 특징은 무선망에서 더 이상 유효하지가 않다. 무선망의 손실률(BER)은 10^{-3} 에서 10^{-2} 정도로 유선망에 비해 상당히 높다. 이 때문에 무선 링크로 전송되는 패킷의 비트 에러로 인한 손실이 빈번하게 발생한다. TCP는 이러한 랜덤 패킷 손실이

* 정 회 원 : 조선대학교 컴퓨터공학부 전임강사
mskang@chosun.ac.kr

** 종신회원 : 조선대학교 컴퓨터 공학부 조교수, 교신저자
sjshin@chosun.ac.kr

[2008/08/20 투고 - 2008/08/25 심사 - 2008/09/09 심사완료]

☆ 이 논문은 2008년도 조선대학교 학술연구비의 지원을 받아 연구되었음

발생하였을 때 네트워크 혼잡이 발생하였다고 오인하고 네트워크 혼잡제어를 실행하여, 전송률을 불필요하게 감소시킨다. 이로 인해 TCP는 무선망이 제공하는 대역폭을 충분히 사용하지 못하게 된다.

이러한 무선 링크의 불안전성을 근본적으로 보완하기 위해 링크 레벨에서 패킷 손실이 발생하면, 링크 레벨에서 자동적으로 패킷을 재전송하는 ARQ (Automatic Retransmission Request)와 같은 메커니즘이 설계되었다. 그러나 ARQ는 손실된 패킷 재전송을 통해, 무선 링크의 패킷 손실을 최소화하고 무선 링크의 안전성을 증대시킬 수 있지만, TCP 수신자의 입장에서는 패킷의 순서가 뒤바뀌는 현상을 발생시킨다.

TCP는 기본적으로 패킷이 순차적으로 전송되는 것을 가정하기 때문에 패킷의 순서가 뒤바뀌어 수신되는 경우, 네트워크 혼잡이 발생하여 패킷 손실이 발생한 것으로 간주하여, Duplicate ACK (중복 응답)을 발생시킨다. 연속된 3개 이상의 Duplicate ACK가 TCP 송신자에게 수신되면, TCP 송신자는 패킷이 손실되었다고 오판하고 손실된 패킷을 재전송함과 동시에 전송률을 떨어뜨려, 혼잡제어를 시도한다. 이는 ARQ의 특성 때문에 실제 패킷 손실이 발생하지 않았으나, 단지 패킷의 순서가 뒤바뀌어 일어나는 현상이다.

이러한 성능 저하 요인을 제거하여 TCP가 무선 네트워크가 제공하는 대역폭을 충분히 사용할 수 있도록 무선 네트워크의 특성이 반영된 TCP로 재설계되어야 한다.

이런 문제를 해결하기 위해 Snoop TCP [4]와 같이 기지국에서 손실된 패킷을 재전송하여 무선 상의 에러를 차단하거나, 무선에러에 의한 패킷 손실과 네트워크 혼잡에 의한 패킷 손실을 구분하여 원인에 따라 다른 처리를 함으로써 성능을 향상시키려는 시도 [2][3]도 있었다.

본 논문은 이미 제안된 여러 가지 무선 TCP 중에서 ABE(Available Bandwidth Estimation: 가용 대역폭 측정) 기법을 이용하여 유/무선상의 패킷 손실에 관계없이 congestion window (혼잡 제어 윈도우)를 결정

하는 TCP-Westwood [1]를 대상으로 한다. TCP-Westwood는 무선 네트워크의 특성중의 하나인 사용자의 이동성 지원(Handoff)에 영향을 받지 않고, 네트워크의 도움을 필요로 하지 않는 점에서 기존의 방법들에 비해 장점이 있으며, 리눅스에 TCP Westwood의 구현이 이미 기본적으로 포함되어 있어, 현재 무선 네트워크상에서 사용자들이 쉽게 사용할 수 있는 유일한 무선 TCP이다. 그러나 본 논문에서 지적하듯이 TCP Westwood는 1% 미만의 낮은 패킷 손실률에서는 기존의 유선을 기반으로 하는 Tahoe, Reno, Sack TCP들에 비해 나은 성능을 보이나, 1% 이상의 높은 패킷 손실률에서는 성능의 향상이 미미하다. 본 논문은 TCP Westwood의 이러한 문제점을 지적하고, 무선 네트워크 상에서 발생할 수 있는 높은 랜덤 패킷 손실에서도 그 성능을 효율적으로 향상시키기 위한 방법을 제안하였다.

본 논문의 구성은 다음과 같다. 2장에서는 무선 TCP의 관련 연구에 대해 기술하고, 3장에서는 시뮬레이션 및 실험을 통한 TCP Westwood의 문제점에 대해 설명한다. 4장에서는 향상된 가용 대역폭 측정 방법에 대한 아이디어를 설명하며, 5장에서는 제시된 아이디어에 대한 성능 평가를 6장에서는 본 논문의 결론 및 향후 연구방향에 대해 기술한다.

2. 관련연구

무선 네트워크상의 TCP의 문제를 해결하고, 무선 네트워크의 대역폭을 충분히 사용하기 위해, 즉 무선 네트워크에서 발생한 패킷 손실이 TCP의 성능에 악영향을 미치지 않게 하기 위해, 여러 가지 방법이 제안되었다. 그 방법은 앞에서 간략하게 언급하였듯이 크게 다음과 같이 세 가지로 나뉜다.

2.1 Split Connection

Snoop TCP [4], Indirect TCP [5]와 같은 Split connection은 그 이름이 의미하는 바와 같이, TCP의 종대중 연결을 송신자와 기지국 사이의 유선 구간,

기지국과 수신자 사이의 무선 구간으로 분리한다. 기지국에는 TCP 패킷 및 ACK의 이동을 감시하는 split connection의 agent가 존재하여, TCP 패킷 전송 상태를 모니터링 한다. 이 agent는 수신자가 보내는 ACK를 통해 무선 구간에서의 패킷 손실을 인지한다. 만약 연속적인 3개 이상의 Duplicate ACK를 발견하면, 이 Agent는 TCP 송신자를 대신하여, 무선 구간에서 패킷이 손실된 것으로 판단하여 그 패킷을 재전송한다. 그리고 Duplicate ACK들을 필터링하여, TCP 송신자가 Duplicate ACK를 수신할 수 없게 한다. 이를 통해 TCP 송신자는 기본적으로 무선 링크에서 발생하는 패킷 손실을 볼 수 없으므로 무선 링크상의 패킷 손실로 인한 네트워크 혼잡 제어를 근본적으로 회피할 수 있으며, 오직 네트워크 혼잡에 의한 패킷 손실에만 반응한다.

이 방법은 기지국의 agent가 무선 링크 상의 패킷 손실을 정확히 인지하고, 그 영향을 제거해주기 때문에 TCP의 성능을 가장 효율적으로 향상시켜 준다. 그러나 agent와 같은 하부 네트워크기능의 도움을 기본적으로 필요로 하기 때문에, 이런 기능을 가지고 있는 무선 네트워크상에서만 사용이 가능하다. 또 사용자의 이동성을 고려하였을 때, 즉 사용자가 하나의 무선 네트워크로부터 다른 무선 네트워크로 이동하였을 때, 기존 무선 네트워크의 agent가 모니터링하고 있는 TCP 세션에 대한 정보(예: 현재까지의 성공적으로 수신된 패킷 시퀀스 넘버)를 다른 무선 네트워크의 agent로 전달하여야 한다. 이러한 문제점들로 인해 널리 사용되기에는 한계가 있다.

2.2 패킷 손실 구분

하부 네트워크의 지원이 필요 없고, 사용자의 이동성을 충분히 지원하기 위해, TCP 송신자와 수신자들만으로 무선 네트워크상에서 성능을 향상시킬 수 있는 방법들이 연구되었다. 기본적 동작은 패킷이 손실되었을 때, 유/무선 네트워크의 전송 특징을 이용하여, 이 패킷 손실이 네트워크 혼잡에 의한

것인지 무선링크에 의한 것인지로 구분하여, 네트워크에 의한 패킷 손실일 경우만 네트워크 혼잡제어를 실행하는 것이다.

패킷 손실 원인을 구분하기 위해 사용되는 유/무선 네트워크의 전송 특징은 Biaz[2]와 Cen[3]에서 언급되었듯이, TCP 송신자가 모니터링 하는 ACK들 간의 도착 시간 간격이다. 네트워크 혼잡이 발생하는 경우, TCP 패킷들의 전송 경로 상에 위치하는 병목 라우터의 큐에 패킷들이 점점 쌓이게 된다. 이런 현상은 패킷들이 TCP 수신자에 도착하는 시간 간격을 점차 증가시키고, TCP 송신자에 수신되는 ACK들의 도착 간격에서도 동일한 패턴이 관찰된다. 그러나 네트워크 혼잡이 발생하지 않는 경우, 패킷이 무선 링크를 경유하더라도 패킷 간의 도착 간격은 증가하지 않는 것이 일반적이다. 따라서 패킷이 손실되었을 경우, 그 직전의 패킷 간 도착시간 간격을 관찰하여 그 간격이 점점 증가하고 있는 상태이면, 패킷 손실의 원인을 네트워크 혼잡으로 추정할 수 있고, 간격의 변화 없이 패킷이 손실되었다면, 무선 링크 상의 랜덤 패킷 손실로 추정할 수 있다. 그리하여, 네트워크 혼잡에 대한 패킷 손실로 판단된 경우에만 네트워크 혼잡제어를 실행하여 성능을 향상시킬 수 있다.

이 방법은 TCP 만으로 성능을 향상시킬 수 있어, 다양한 무선 네트워크상에서 사용될 수 있다는 장점이 있으나, 네트워크 혼잡으로 추정할 수 있는 패킷 전송 지연 기준(어디까지 지연되었을 때 네트워크 혼잡으로 판단할 것인지)을 정확히 추정하기 어렵다는 문제가 있다. 즉, 패킷 전송 지연 기준의 설정에 의해 패킷 손실의 원인 판단의 정확도가 달라진다. 패킷 전송 지연 기준을 너무 작게 잡게 되면, 네트워크 혼잡으로 인한 패킷 손실로 판단할 확률이 높아지게 되고, 그 결과 불필요한 네트워크 혼잡제어를 수행하게 되어 네트워크에서 제공하는 대역폭을 충분히 사용할 수 없게 된다. 또 패킷 전송 지연 기준을 너무 크게 잡게 되면, 무선 링크에서의 오류에 의한 패킷 손실로 판단할 확률이 높아지며, 이 경우 실제 네트워크 혼잡으로 발생한 패

킷 손실에 대해 적절한 네트워크 혼잡제어를 못하게 되어 네트워크 혼잡을 가중시키게 된다. 패킷 손실 구분 방법의 정확도는 적절한 패킷 전송 지연 기준의 설정에 달려 있으나 이 부분을 정확히 결정하기 어렵다는 단점이 있다.

2.3 가용대역폭 측정

패킷 손실 구분의 정확도가 떨어질 수 있다는 단점을 회피하면서 하부 네트워크의 도움 없이 TCP Westwood [1]와 같이 TCP 자체만의 수정으로 무선 네트워크에서 성능을 향상시키기 위해 가용대역폭을 측정하여 사용하는 방법이 제안되었다. 이 방법은 TCP가 자신이 사용할 수 있는 네트워크 평균 가용대역폭을 지속적으로 측정하여, 패킷 손실이 발생했을 경우, 그것이 네트워크 혼잡으로 인한 패킷의 손실인지 무선링크의 에러에 의한 손실인지 구분하지 않고, 혼잡조절 윈도우 (cwnd: congestion window)의 크기를 측정된 대역폭으로 설정하여 주는 방법이다. 그러므로 패킷 손실을 구분할 필요가 없어 유/무선 네트워크에 상관없이 사용할 수 있다는 장점이 있다.

TCP Westwood의 가용대역폭을 측정하는 알고리즘은 패킷 페어(packet pair) 알고리즘[6]에 기반을 두고 있다. 패킷 페어란 동시에 보내는 두 개의 패킷을 의미한다. 병목현상이 발생하는 라우터에 의해 두 패킷의 도착 시간에 차이가 발생하게 되고 이 시간을 이용하여 대역폭을 측정하는 것이 패킷 페어 알고리즘의 기본 아이디어이다.

TCP Westwood는 ACK를 매번 수신할 때마다 다음의 가용 대역폭 측정 공식을 사용한다.

$$B = ACK_i / (T_i - T_{i-1})$$

$$ABE = (1 - \alpha)ABE + \alpha B$$

ACK_i 는 i 번째 ACK가 acking(도착응답)하는 데이터 바이트를 나타낸다. T_i 는 i 번째 ACK가 도착한 시간을 나타낸다. a 는 moving average의 파라미터를

나타내며, ABE 는 a 와 B 에 의한 moving average로 나타난다. 이 ABE 가 TCP Westwood가 측정한 평균 가용 대역폭이 된다.

- (1) Upon a packet loss
 $cwnd \leftarrow ABE, ssthresh \leftarrow ABE$
- (2) Upon a timeout
 $cwnd \leftarrow 1, ssthresh \leftarrow ABE$

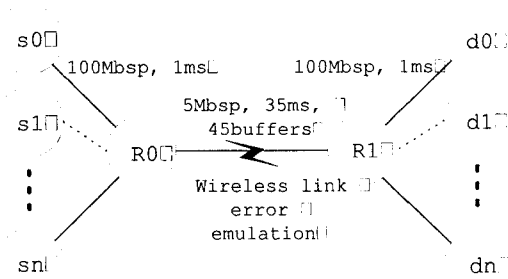
TCP Westwood는 패킷 손실 시 혼잡제어 윈도우 (cwnd)와 ssthresh(slow start threshold)을 측정된 가용 대역폭 값, ABE 로 설정한다. 이것이 기존의 Reno와 같은 유선 기반의 TCP에서 무조건적으로 cwnd를 $1/2 \cdot cwnd$ 로 감소시키는 것과 다른 점이다. timeout의 경우엔 cwnd 값은 1로 설정되고 ssthresh는 ABE 로 설정된다. 이를 통해 TCP Westwood는 패킷 손실 발생 원인에 관계없이 유/무선 네트워크가 제공할 수 있는 대역폭을 사용할 수 있게 된다.

TCP Westwood가 무선 네트워크의 대역폭을 충분히 사용하려면, 대역폭을 정확히 측정할 수 있어야 한다는 점에 유의해야 한다. 그러나 무선 네트워크에서의 성능 향상을 위해 개발된 TCP Westwood가 무선 네트워크상의 랜덤 패킷손실이 1%(패킷 기준)이상이 될 경우 ABE 측정의 정확도가 심각하게 왜곡되는 현상이 발생한다. 이 부정확한 값은 TCP Westwood가 무선 네트워크가 제공하는 대역폭을 충분히 사용하지 못하게 한다.

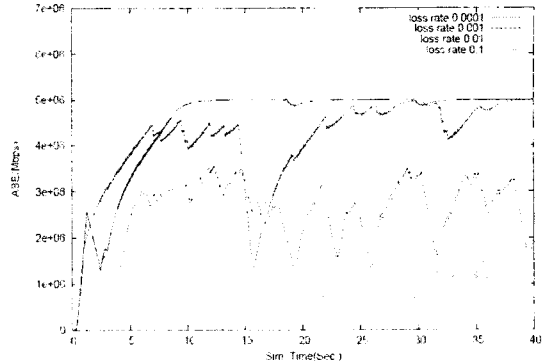
3. TCP Westwood의 ABE 분석

랜덤 패킷 손실이 TCP Westwood의 ABE 측정에 미치는 영향을 분석하기 위해 네트워크 시뮬레이터 (NS2)[7]를 이용하여 다음과 같은 시뮬레이션을 실행하였다.

그림 1은 시뮬레이션에 사용된 네트워크 모델이다. R0와 R1은 5Mbps의 링크 용량을 가지고 있고 35ms의 delay를 가진다. R0에서 확률 p (랜덤 패킷



(그림 1) 시뮬레이션 네트워크 토폴로지



(그림 2) 랜덤패킷손실률에 따른 ABE

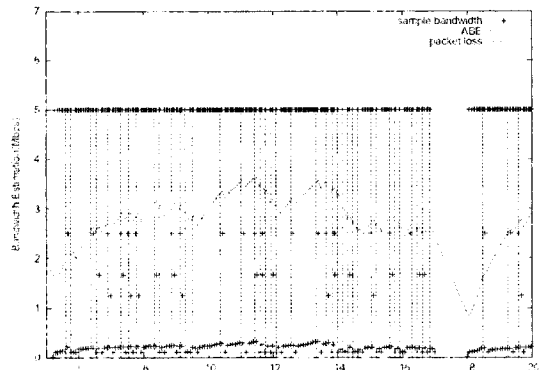
손실률)로 패킷을 버림으로써 wireless link error를 에뮬레이션하고 있다. sn과 dn은 송신자와 수신자를 의미한다. 패킷의 크기는 1000byte로 설정하였고 40초 동안 시뮬레이션을 실행하였다.

3.1 랜덤 패킷 손실률에 따른 ABE 변화

그림 2는 랜덤 패킷 손실률의 변화에 따라 하나의 TCP Westwood 송/수신자의 ABE 측정값을 보여주는 그래프이다. 이 그림에서 패킷 손실률이 0.01%인 경우, 즉 순수한 유선 네트워크인 경우 ABE의 값은 거의 5Mbps로 정확하게 병목 구간의 대역폭을 측정하고 있다(하나의 TCP 연결이 병목 구간의 대역폭을 모두 사용하므로). 그러나 랜덤 패킷 손실률이 0.1%로 증가하게 되면, 그 정확도가 떨어지는 것을 알 수 있다. 패킷 손실률이 1%로 더욱 증가하게 되면, ABE의 값은 5Mbps로 측정되지 못하고 심각하게 왜곡되는 현상을 보여 주고 있다. 패킷 손실률 1%는 무선 링크가 일반적으로 고려하고 있는 최대 패킷의 손실률이기 때문에 1%까지는 TCP Westwood의 ABE측정이 정확하여야 무선 네트워크의 대역폭을 효율적으로 사용할 수 있다. 그러나 이 그래프에서 알 수 있듯이 1%의 패킷 손실률에서 ABE의 측정이 부정확해지고, 실제 값보다 낮은 값을 가지게 됨으로써, TCP의 성능이 저하되는 문제점을 가지게 된다. 패킷 손실률이 10%인 경우는 ABE측정 자체가 불가능해진다는 것을 알 수 있다.

3.2 랜덤 패킷 손실률에 따른 ABE 변화

ABE는 2.3절에 언급되었듯이, 매 ACK가 도착할 때마다 측정되어지는 B(샘플 ABE)의 평균값이다. 부정확한 ABE가 측정되었다는 것은 이 B값이 부정확하게 계산되는 빈도수가 많다는 것을 의미한다. 랜덤 패킷 손실이 ABE 측정에 미치는 영향을 분석하기 위해, 이 B값이 어떻게 측정이 되고, 어떤 이유에 의해 부정확하게 되는지를 분석하였다. 그림 3은 패킷 손실률 1%의 ABE값 측정에서 시뮬레이션 시간 0에서 20초 동안 해당 ABE값에 대응되는 B값 및 패킷 손실이 발생한 시각을 같이 그린 것이다. '+' 마크가 B값을 나타내며, 수직 점선이 패킷 손실이 발생한 시각을 나타낸다.



(그림 3) 패킷 손실률 1%에서의 ABE 측정에 영향을 미치는 B값 및 패킷 손실의 연관 관계

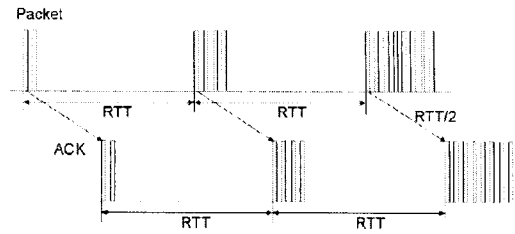
이 그림을 보면 패킷 손실이 발생하지 않으면 ABE값이 점차 증가하나, 12초 직전 근처의 구간이나 14초 직후 근처의 구간에서 볼 수 있듯이, 패킷 손실이 발생하면 그 값이 감소한다는 사실을 알 수 있다. 이것은 패킷 손실이 없는 정상적인 상태인 경우, B값이 5Mbps로 계산이 되는 경우가 상대적으로 많이 발생(10초 근방 위 부분)함을 의미한다. 물론, 패킷 손실이 발생하지 않더라도 B값이 0근처로 측정되는 경우(10초 근방 아래 부분)도 있으나 그 빈도수는 5Mbps로 측정되는 경우보다 상대적으로 작다. 그러므로 평균값 ABE는 점차 증가를 하게 된다. 이와 반대로, 패킷 손실이 발생한 경우, 거의 0에 가깝게 측정이 되는 경우가 상대적으로 많이 발생(12초 직전 근방의 위와 아래를 비교)하는 것을 관찰할 수 있다. 이 경우 평균 ABE값은 점차 감소하게 된다.

패킷 손실이 발생하는 경우 왜 B값은 0에 가까운 값을 가지게 되는 것이고, 패킷 손실이 발생하지 않는 경우는 B는 왜 5Mbps의 값을 가지게 되는 것일까? 이 의문에 대한 답은 패킷 페어 알고리즘의 정의, 2.3절의 B의 계산식 및 패킷 손실 발생시 TCP의 패킷 송수신의 패턴을 종합적으로 고려하였을 때 찾을 수 있다.

그림 4는 패킷 손실이 발생하지 않을 때 TCP 송신자가 어떠한 패턴으로 패킷을 전송(상단부)하고, TCP 수신자가 어떤 패턴으로 ACK를 수신하는 지(하단부)를 시간의 흐름에 따라 보여주고 있다.

TCP 송신자가 처음 두 개의 패킷을 송신했을 때 (cwnd=2), 그 패킷들이 손실되지 않고 성공적으로 TCP 수신자에게 도착하면, TCP 수신자는 ACK를 보내어 패킷들의 도착에 대해 응답한다. TCP 송신자가 이 ACK들을 수신하게 되면, cwnd를 4로 증가시키고(slow start이기 때문에) 4개의 패킷을 전송하게 된다. 이와 같이 TCP 송신자는 ACK의 도착에 맞추어 새로운 패킷을 전송하기 때문에 TCP 송신자는 매 RTT (Round Trip Time)의 간격으로 새로운 패킷들을 촘촘하게 송신하게 된다.

이런 TCP의 버스티한 트래픽 특징은 패킷 페어



(그림 4) TCP의 패킷 송신 패턴 및 ACK의 도착 패턴

알고리즘 및 B의 계산에 부정확한 요인을 포함시킨다. 패킷 페어 알고리즘에서는 두 개의 패킷을 동시에 보내어 두 개의 패킷이 얼마만큼 떨어져서 도착하는 지를 측정한다. B의 계산에서 i 번째 ACK와 $i-1$ 번째 ACK의 도착시간의 차이가 이 시간 간격에 해당한다. 그러므로 정확하게 B를 계산하기 위해서는 두 개의 패킷이 동시 또는 거의 인접하여 전송되어야 한다는 전제조건이 필요하다. TCP 트래픽의 버스티한 특징으로 인해 이러한 전제조건이 유효하지 않은 경우가 발생한다. 하나의 주기 내에 송신되는 패킷들은 서로 인접하여, 거의 동시에 송신되어 패킷페어의 전제조건을 만족하지만, 주기의 경계에 위치하는 패킷들은 처음부터 RTT의 간격으로 송신된다. 이 경우, ACK들의 도착 간격이 RTT 이상의 값으로 본래의 패킷 페어의 전제조건을 만족시키지 못한다. B값을 계산할 때 이런 잘못된 패킷페어에 대한 구분을 하지 않기 때문에 정확하지 않은 B값들이 발생하여 ABE 측정에 악영향을 미치게 된다. 이런 예는 그림 3의 10초 근방에서 패킷 손실이 발생하지 않았던 구간에도 B값이 0근처로 측정되는 경우에서 다수 확인할 수 있다.

패킷 손실이 발생하지 않는 경우에는, 이러한 오측정이 발생하긴 하지만, cwnd의 값이 충분히 커지게 되면, 즉 TCP 송수신자가 사용할 수 있는 대역폭이 패킷들로 충분히 차게 될 경우, 정상적인 B값 측정이 비정상적인 B값의 측정에 비해 그 값이 현저하게 많아지고, 최종적인 ABE의 결과가 정상적인 값에 수렴을 하지만, 패킷 손실이 발생하는 경우는 cwnd의 값이 충분히 커지지 못하기 때문에,

비정상적인 B값이 정상적인 B값에 비해 많아지게 되고, 그 결과, ABE가 정확하지 못한 값으로 측정 된다.

결국, 현재 TCP Westwood의 ABE측정은 패킷 손실이 빈번하면 할수록 그 정확도가 떨어지는 문제점을 가지고 있고, 무선 네트워크를 위해 TCP Westwood가 디자인 되었다는 점을 고려할 때, 어느 정도의 랜덤 패킷 손실에도 정확한 ABE측정이 가능해야 함에도, 부정확한 값을 계산함으로써 무선 네트워크가 제공하는 대역폭을 충분히 사용하지 못하게 된다.

4. 새로운 ABE 제안

정확한 ABE를 측정하기 위해서는 잘못된 B값을 ABE 계산에 포함 시키지 말고 걸러 내야 한다. 즉, 동시에 보내지는 패킷에 대해서만(패킷 페어 알고리즘의 정의에 따라) B값 및 ABE 계산을 수행하고, 그 외의 경우 동시에 보내어지지 않는 패킷 페어(예: 버스티한 트래픽의 경계에 위치하고 있는 패킷 쌍)에 대해서는 B값 및 ABE 계산에서 제외를 시켜야 한다.

변형 알고리즘 1

*i*번째 ACK가 도착한 경우,

if $S_n - S_{n-1} > S_{thresh}$ return;

else

$$B = ACK_i / (T_i - T_{i-1})$$

$$ABE = (1 - \alpha)ABE + \alpha B$$

이를 위해 변형 알고리즘 1을 제안하였다. *i*번째 ACK가 도착한 경우, 이 ACK를 유발한 *n*번째 패킷의 출발 시간인 S_n 과 그 직전 전송 패킷인 *n-1*번째 패킷의 출발시간인 S_{n-1} 의 차이를 계산하여, 정의된 특정시간(S_{thresh}) 보다 큰 경우, B 및 ABE 계산을 수행하지 않고, S_{thresh} 의 범위에 들 경우만 B 및 ABE를 계산함으로써 잘못된 B의 값을 걸러낼 수 있게 한다. 이 방법은 잘못된 B의 값을 정확히 걸

러 낼 수 있지만, 각 패킷의 전송시간을 정확히 기억해야 된다는 점에서 메모리의 사용을 증대시키고 *i*번째 ACK에 대응되는 *n*번째 패킷의 출발시간을 정확히 매핑해야 하기 때문에 구현 코드가 복잡해지는 단점이 있다. 또 S_{thresh} 의 값에 따라 필터링 되는 B의 개수가 달라지기 때문에 정확한 ABE의 값을 측정할 수 있는 S_{thresh} 의 값을 결정하여야 한다. 이 논문에서는 S_{thresh} 를 $1/16 \cdot RTT$ 로 정의하였다. 패킷 손실률과 S_{thresh} 의 상관관계는 추후 연구로 진행할 계획이다.

변형 알고리즘의 문제점을 보완하기 위해 다음의 변형 알고리즘 2를 제안하였다. 변형 알고리즘 2는 잘못된 B를 걸러낸다는 점에서 변형 알고리즘 1과 동일하나 메모리 사용 및 코드 구현량의 문제를 해결하기 위해 *n*번째 전송 패킷의 전송시간을 사용하지 않는다. 대신, ACK의 도착 간격이 패킷의 전송 간격과 유사하다는 점에 착안하여(그림 4 참조), ACK의 도착 시간의 평균값 및 분산을 이용하여 RTT계산식과 동일하게 B값 필터링을 위한 threshold 값을 다음과 같이 결정한다.

변형 알고리즘 2

*i*번째 ACK가 도착한 경우,

$$T = T_i - T_{i-1} \quad \text{--- (1)}$$

$$T_{average} = (1 - g) \cdot T_{average} + g \cdot T \quad \text{--- (2)}$$

$$T_d = (1 - h) \cdot T_d + h \cdot |T_{average} - T| \quad \text{--- (3)}$$

$$T_{thresh} = T_{average} + 3 T_d \quad \text{--- (4)}$$

if $T > T_{thresh}$ return;

else

$$B = ACK_i / (T_i - T_{i-1})$$

$$ABE = (1 - \alpha)ABE + \alpha B$$

식(1)은 *i*번째 ACK의 도착 간격을 의미하며, 식(2)는 이러한 도착 간격의 평균값을 moving average를 이용하여 계산한다. 식(3)은 도착 간격의 표준편차를 계산하고, 계산된 평균과 표준편차를 이용하여, 식(4)를 계산, B값 필터링을 위한 기준값 T_{thresh} 로 사용한다. ACK의 도착 간격 T가 T_{thresh} 보

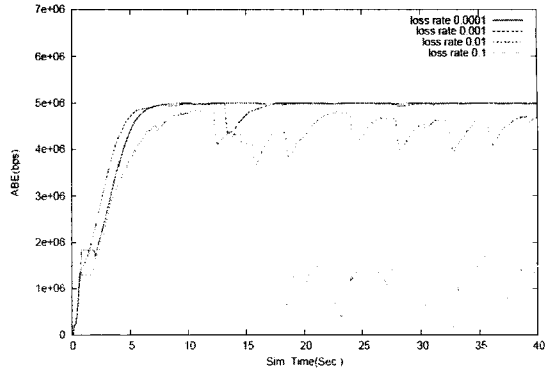
다 작을 경우에만 적절한 B로 추정하고 ABE 계산에 사용한다.

이런 방법을 사용하여, 메모리 사용량 및 코드 구현량을 줄이면서, 부정확한 B값을 걸러낼 수 있다는 장점이 있으나, 이 방법 역시 T_{thresh} 의 값에 따라 정확도가 달라진다. 특히 T_{thresh} 를 계산할 때 사용되는 T도 각 주기사이의 RTT값이 포함될 수 있으므로(3.2절에서 언급되었던 동일한 문제점을 야기), 패킷 손실률이 증가하면 T_{thresh} 의 값이 ABE의 값처럼 부정확하게 커질 가능성이 존재한다. 이런 추론은 시뮬레이션을 통한 성능 평가 결과에서 확인할 수 있다. 시뮬레이션에서 g와 h의 값은 RTT를 계산하는 데 사용되는 값과 동일하게 각각 0.25와 0.125로 사용하였다.

5. 시뮬레이션을 통한 성능 평가

제안된 알고리즘의 성능을 3절에서 설명한 시뮬레이션 환경과 동일하게 설정하여 평가하였다(그림 1 참조). 성능 평가의 지표는 3절과 동일하게 ABE의 정확성으로 하나의 TCP Westwood 흐름에 대해 패킷 손실률에 따른 ABE를 측정하였다.

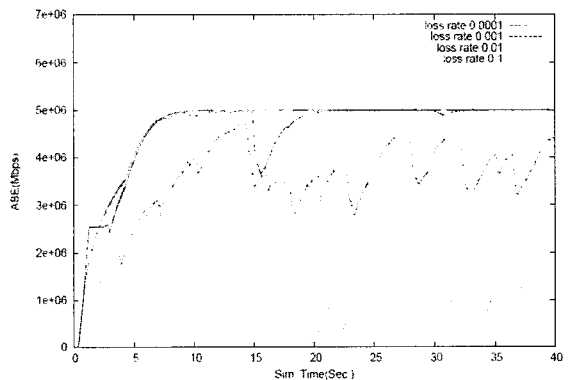
그림 5는 변형 알고리즘 1의 ABE 측정값을 시뮬레이션 시간에 따라 그린 것이다. 그림 2와 이 그래프를 비교해보면 변형 알고리즘이 기존 ABE 측정에 비해 정확하다는 사실을 알 수 있다. 패킷 손실률이 0.01%인 경우에 기존 ABE와 같이 병목 구간의 5Mbps를 정확히 측정하며, 패킷 손실률이 0.1%인 경우 ABE 값이 거의 5Mbps로 기존 방법에 비해 훨씬 정확한 ABE 값을 보여준다. 패킷 손실률이 1%인 경우, 그 차이를 확연히 확인할 수 있다. 그림 2의 경우, ABE의 값이 3Mbps이하의 값으로 측정되며 ABE의 측정이 안정적이지 못하고, 들쭉날쭉 하였다. 그러나 변형 알고리즘 1의 경우, 정확히 5Mbps로 측정하지는 못하나, 그 값이 평균 4.5Mbps 정도로 측정되며, 들쭉날쭉한 정도가 훨씬 적다. 그러므로 동일한 랜덤 패킷 손실률에서, 변형 알고리즘 1을 사용하면 TCP Westwood가 네트워크



(그림 5) 변형 알고리즘 1의 ABE 측정

의 대역폭을 효율적으로 사용할 수 있다. 패킷 손실률이 10%인 경우는 TCP 자체가 패킷 전송을 정상적으로 할 수 없으므로, ABE 측정에 필요한 ACK의 도착 빈도가 충분하지 못하다. 그러므로 ABE의 측정이 가능하지 않게 된다.

그림 6은 변형 알고리즘 2의 ABE 측정값을 시간에 따라 그린 것이다. 그림 2, 5과 비교를 하면, 변형 알고리즘 2의 특징을 알 수 있다. 변형 알고리즘 2는 변형 알고리즘 1에 비해, 메모리의 사용과 코드 구현량이 적다는 이점이 있다. 그러나 잘못된 B값을 필터링 하기 위해 사용되는 threshold 값이 패킷 손실률에 따라 그 정확도가 달라진다는 단점이 있었다. 그림 6에서 패킷 손실률이 0.01%인 경우 변형 알고리즘 2도 정확히 5Mbps의 ABE값을 측정하고 있다. 패킷 손실률이 0.1%인 경우, 변형



(그림 6) 변형 알고리즘 2의 ABE 측정

알고리즘 2의 ABE 측정 정확도는 그림 2에서 보여주는 기존의 ABE 측정에 비해서는 상당히 정확하고 안정적이라는 것을 알 수 있다. 또 그림 5에서 보여주는 변형 알고리즘 1의 ABE 정확도와 유사함을 볼 수 있다. 그러나 패킷 손실률이 1%가 되는 경우, 변형 알고리즘 2의 ABE는 그림 2의 ABE 측정값에 비해서는 안정적이거나 그림 5의 변형 알고리즘 1의 ABE와 비교해 보면 정확도가 많이 떨어짐을 알 수 있다. 정확도가 떨어진다는 것은 잘못된 B값을 정확히 필터링을 못해준다는 것을 의미하고 이것은 T_{thresh} 가 정확한 값을 갖지 못하는 것을 의미한다. 이것은 패킷 손실률이 증가함에 따라 $cwnd$ 가 충분히 커지지 못하고 정상적인 패킷들을 촘촘히 보내지 못해, 버스티한 패킷 전송주기가 자주 반복되어(예: 그림 4) T_{thresh} 계산에 포함되지 말아야 할 ACK들 간의 도착 간격(버스티 패킷 전송들 간의 간격인 RTT)이 자주 포함된다는 것을 의미한다. 이때 T_{thresh} 값이 상대적으로 커져, 잘못된 B값이 ABE 값에 자주 포함되어 ABE의 정확도가 떨어지게 된다. 패킷 손실률이 10%인 경우, 그림 2, 5와 동일하게 ABE 측정이 불가하게 된다.

6. 결론 및 향후 연구

본 논문에서는 무선 TCP 종류 중의 하나의 TCP Westwood가 무선 네트워크상의 랜덤 패킷 손실률의 증대에 따라 성능이 저하되는 요인을 분석하였다. 랜덤 패킷 손실률이 증가할수록 TCP Westwood의 가용 대역폭 측정(ABE)의 정확도가 떨어지는 원인을 시뮬레이션 분석을 통해 그것이 잘못된 패킷 페어 계산에 있음을 밝혔다. 이런 원인 분석에 따라 잘못된 B값을 제거할 수 있는 두 개의 알고리즘들을 제안하였으며, 시뮬레이션을 통해 제안된 알고리즘에서 ABE 측정의 정확도가 향상되었음을 보였다. 향상된 ABE 측정 기법에서 정확도를 결정하는 S_{thresh} 와 T_{thresh} 의 영향이 패킷 손실률과 밀접한 관계가 있음을 시뮬레이션을 통해 인지하였으나 그것의 상관관계를 정확하게 해석하는 것은 향후 연

구에서 진행할 예정이며, 이런 부분들을 리눅스 시스템에 구현하여 실제 실험을 통해 분석할 예정이다.

참고 문헌

- [1] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi, and R. Wang, "TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links," In Proceedings of ACM Mobicom 2001, pp 287-297, Rome, Italy, July 16-21 2001.
- [2] Saad Biaz and Nitin H. Vaidya, "Discriminating Congestion Losses from Wireless Losses Using Inter-Arrival Times at the Receiver," Proceedings of the 1999 IEEE Symposium on Application - Specific Systems and Software Engineering and Technology, 1999.
- [3] Song Cen, Pamela C. Cosman, Geoffrey M. Voelker: End-to-end differentiation of congestion and wireless losses. IEEE/ACM Transactions on Networking. 11(5): 703-717 (2003).
- [4] H. Balakrishnam, S. Seshan, E. Amir, and R. Katz, "Improving TCP/IP performance over wireless networks," In First ACM International Conference on Mobile Computing and Networking(MOBICOM), November, 1995.
- [5] A. Bakre and B. R. Badrinath, "Indirect TCP for Mobile Host," 15th International Conference on Distributed Computing Systems, 1995.
- [6] S. Keshav, "A Control-Theoretic Approach to Flow Control," Proc.ACM SIGCOMM, pp. 3-15, Sept. 1991.
- [7] UCB/LBNL/VINT Network Simulator, URL:<http://www.isi.edu/ns-nam/ns>.

○ 저 자 소개 ○



강 문 수(Moonsoo Kang)

1998년 한국과학기술원(KAIST) 전산학과(학사)
2000년 한국정보통신대학교(ICU) 공학부(석사)
2007년 한국정보통신대학교(ICU) 공학부(박사)
2007년~현재 조선대학교 컴퓨터공학부 전임강사
관심분야 : Congestion Control, VANET, 병렬시스템 등.
E-mail : mskang@chosun.ac.kr



신 석 주(Seokjoo Shin)

1997년 한국항공대학교 항공전자공학과 졸업(학사)
1999년 광주과학기술원 정보통신공학과 졸업(석사)
2002년 광주과학기술원 정보통신공학과 졸업(박사)
2002~2003 한국전자통신연구원 이동통신연구단 선임연구원
2003~현재 조선대학교 컴퓨터공학부 조교수
관심분야 : 무선통신 MAC 기술, 센서 네트워킹 기술, etc.
E-mail : sjshin@chosun.ac.kr