

# 초고속 네트워크에서 동적 가용대역폭 획득 기반 TCP 전송기법

박민우\*, 추현승\*

\*성균관대학교 정보통신공학부  
e-mail:{minustyle, choo}@skku.edu

## TCP Congestion Control based on Dynamic Available Bandwidth Estimation for High-Speed Networks

Minu Park\*, Hyunseung Choo\*

\*School of Information and Communication Engineering,  
Sungkyunkwan University

### 요 약

무선 통신 기술의 발전과 함께 802.11n과 같은 광대역 네트워크 인프라의 구축이 확대됨에 따라, 이에 적합한 높은 성능을 낼 수 있는 전송 계층의 프로토콜이 필요하다. 유선 환경에서 개발된 TCP(Transmission Control Protocol)는 현재 가장 널리 이용하는 전송 프로토콜이다. 하지만 높은 에러율과 긴 딜레이가 발생하는 무선 네트워크상에서는 패킷 손실의 원인을 구별하지 못하기 때문에 성능이 저하된다. 또한, 광대역폭을 지원하는 초고속 네트워크상에서 TCP가 동작할 경우 기존 무선 네트워크에서 처리하는 혼잡윈도우 값보다 커진 값을 제대로 처리하지 못하기 때문에 가용대역폭을 활용하지 못하는 문제점이 발생한다. 따라서, 본 논문에서는 초고속 네트워크 안에서 slowstart threshold에 기반하여 가용대역폭을 동적으로 획득하고, 높은 전송률을 보장하는 TCP VEST(TCP Vegas-based Estimator with Slowstart Threshold)를 제안한다. TCP VEST는 기본적으로 송신자기반 무선네트워크에서 우수한 성능을 지원하는 TCP WestwoodVT의 패킷 손실원인 구분법을 사용한다. 그리고 TCP VEST는 패킷 손실의 원인을 구분하고 각각의 경우에 따라서 전송량을 slowstart threshold 값의 변화에 맞춰 조절한다. NS-2를 통한 성능평가에서 VEST는 WestwoodVT와 비교했을 때 링크 에러율이 1%인 환경에서는 20% 이상의 전송률 향상을 나타내며, Westwood와 비교했을 때는 60%의 성능향상이 있다. 또한 초고속 네트워크에서 5%와 10% 사이의 에러율을 가질 때에는 WestwoodVT와는 50%, Westwood와는 300% 이상의 성능향상이 나타남을 보인다. 이를 통해 VEST가 기존의 무선 네트워크와 더불어 초고속 네트워크에서도 동적으로 대역폭을 획득함으로써 높은 전송률을 지원하는 것을 입증한다.

### 1. 서론

TCP(Transmission Control Protocol)[1]는 전송계층에서 가장 널리 사용되는 프로토콜로써, 신뢰성있는 전송을 보장하기 위해 혼잡회피 알고리즘과 혼잡제어 알고리즘[2]을 사용한다. 그러나 네트워크의 형태가 유선에서 무선, 그리고 최근에는 광대역 네트워크로 확대되면서 보다 안정적이고 빠른 전송을 위하여 기존 TCP 기법의 수정은 불가피하다. 따라서 무선 TCP 기법들은 무선 네트워크 환경이 갖는 신호 손실, 신호 감쇄 등과 같은 문제점을 가용 대역폭 측정 및 패킷 손실의 원인 규명 후 처리 등과 같은 기법을 통해 전송률을 향상시킨다[3].

그렇지만, 기존의 무선 TCP 기법들은 초고속 무선 네트워크 환경에서 동작할 때, 수신자가 이동통신기나 무선 인터넷을 사용하면서 통신하는 양이 급속하게 증가하기 때문에 기존에는 고려하지 않았던 큰 대역폭의 비효율적인 사용과 같은 문제점을 경험한다. 기본적으로 TCP는 패킷이 손실되기 전까지는 전송량을 산술적으로 증가하고 문제가 발생하면 전송량을 반으로 줄이는 AIMD(Additive

Increase, Multiplicative Decrease)[4] 알고리즘을 통해 전송을 유지한다. 하지만, 초고속 네트워크에서는 기존에 고려했던 전송윈도우 값보다 훨씬 큰 혼잡윈도우 값을 가지게 된다. 따라서 TCP가 패킷 손실이 발생한 후 빠른 회복 모드(Fast Recovery)를 동작할 때 큰 윈도우 값으로 복구하는데 많은 시간이 걸리고 네트워크에서 제공하는 큰 대역폭을 제대로 활용하지 못하는 문제점이 있다. 결과적으로 현존하는 무선 TCP 기법들은 초고속 네트워크 안에서 동작할 때 낮은 성능을 나타낸다.

본 논문에서는 초고속 네트워크 안에서 slowstart threshold에 기반하여 가용대역폭을 동적으로 획득하고, TCP NewReno, TCP Westwood, WestwoodVT[5]와 비교했을 때 높은 성능을 내는 TCP VEST(Vegas-based Estimator with Slowstart Threshold)를 제안한다. TCP VEST는 우선 WestwoodVT의 Vegas 기반 네트워크 버퍼상태 확인 알고리즘을 사용하여 패킷 손실의 원인을 링크 에러, 네트워크 혼잡, 보류 상태로 구분한다. 그 후 패킷 손실의 원인에 따라 slowstart Threshold 값의 변화를

살피고 혼잡윈도우 값을 수정하여 초고속 네트워크에서 패킷 손실이 발생한 경우에도 동적으로 대역폭을 획득할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 무선 네트워크 환경에서 동작하는 기존의 무선 TCP 연구를 소개하고, 3장에서는 제안하는 알고리즘을 설명한다. 4장에서는 시뮬레이션을 통해 제안하는 알고리즘의 성능을 비교 분석하고 최종적으로 5장에서는 논문의 결론을 맺는다.

## 2. 관련연구

본 장에서는 기존 무선 네트워크에서 동작하는 무선 TCP 기법들을 설명하고, 패킷 손실이 발생했을 때 전송량 조절 방법[6]을 비교한다. Original TCP를 무선에 적용시키기 위해 제안된 TCP NewReno를 시작으로 현재까지 무선 네트워크에서 우수한 성능을 내는 TCP Westwood, TCP VEST의 모태 알고리즘인 WestwoodVT에 대해 설명한다.

### 2.1 TCP New Reno

현재까지 널리 사용되는 표준 TCP인 TCP Reno는 단일 패킷 손실의 경우만을 고려했기 때문에 무선 환경에서 발생할 수 있는 연속적인 패킷 손실 상황에서 송신자의 혼잡윈도우를 연속해서 반으로 줄여나가므로 성능이 좋지 않은 단점이 있다. 이 단점을 극복하고자 제안된 새로운 전송 기법인 TCP New Reno는 기본적으로 빠른 회복(Fast Recovery) 알고리즘인 TCP Reno의 알고리즘과 동일하며, 패킷 손실이 발생된 경우에 빠른 재전송(Fast Retransmit) 알고리즘을 수행한다. 패킷 손실 발생 후 3개의 중복 ACK를 받으면 빠른 회복 알고리즘을 수행하는데 이 때, 부분 ACK(Partial ACK)로 판단이 될 경우에는 다중으로 패킷 손실 발생한 것으로 가정한다. 따라서 계속 빠른 회복 알고리즘 상태를 유지하면서 손실된 패킷을 재전송한 후 패킷 손실 발생 시점 이후에 보낸 데이터에 대한 ACK를 받으면 빠른 회복 알고리즘을 종료한다. 즉, 다중 패킷 손실이 발생하여도 전송량은 하나의 패킷손실이 일어난 것과 동일한 감소량을 보인다.

그러나 무선 네트워크에서 TCP New Reno는 네트워크 혼잡에 의한 패킷 손실과 BER(Bit Error Rate)에 의한 패킷 손실을 구별하지 못하기 때문에 빠른 회복 알고리즘을 효과적으로 수행하지 못한다.

### 2.2 TCP Westwood

TCP Westwood는 무선 네트워크 환경에서 우수한 성능을 제공하는 TCP 스킴으로 송신자가 받은 ACK 패킷 배율을 기반으로 현재 보낼 수 있는 전송량을 예측한다. 패킷 손실 발생시 이렇게 예측된 전송량을 기반으로 손실된 패킷을 재전송할 때 전송량을 설정한다. TCP Westwood는 무선 네트워크 환경의 TCP 스킴중 가장 빠

른 전송률을 제공한다. 그러나 패킷 손실의 원인에 대한 판단이 없기에 패킷 손실이 발생할 때 마다 전송량을 조절하는 단점이 있다. 따라서 링크의 에러율이 높아지면 전반적인 네트워크 성능이 떨어지게 되어 전송률이 낮아지게 된다. 또한, 전송중인 트래픽 정도에 따라 가용 대역폭 예측의 정확도 역시 같이 떨어지기 때문에 항상 높은 전송률을 보장하지 못하는 문제점이 있다.

## 2.3 TCP WestwoodVT

TCP WestwoodVT는 무선 네트워크에서 발생하는 패킷 손실의 원인을 판단하지 못하는 TCP Westwood의 문제점을 보완한 방법으로, 3-dupack 발생 원인을 규명하기 위해 네트워크 노드들의 버퍼상태를 이용하며 상태를 확인하는 방법이다. TCP Vegas의 아이디어를 활용하여 ack를 받을 때 마다 현재 네트워크 상태에서 보낼 수 있는 최고의 예상 전송률과 현재 보낸 실제 전송률을 계산한다. 이 두 값의 차이를  $\Delta$  변수로 표시하고, 이 값은 네트워크 노드 버퍼에 존재하는 데이터의 양으로 파악할 수 있다.

WestwoodVT는 3-dupack을 받을 때  $\Delta$ 의 값을 계산한 후 버퍼 한계점인  $\alpha$ ,  $\beta$ 와 비교한다. 만약  $\Delta > \beta$ 인 경우에는 네트워크 노드들이 허용할 수 있는 버퍼 한계점 이상으로 데이터가 존재함을 의미하고, 패킷 손실의 원인을 네트워크 혼잡으로 판단한다. 그리고 Westwood의 Bandwidth Estimation을 이용해서 혼잡제어 알고리즘을 수행하고 손실된 데이터 패킷을 재전송한다.  $\Delta < \alpha$ 인 경우는 네트워크 노드들의 버퍼가 비워져 있음을 의미하고 이때 발생한 3-dupack의 원인은 비트 에러로 판단 가능하다. 이 때는 혼잡제어 알고리즘을 수행하지 않고 cwnd를 하나 증가시킨 값으로 재전송을 수행한다. 마지막 경우인  $\alpha < \Delta < \beta$ 에서는 3-dupack 원인 판단을 규명하지 않는다. 즉, 현재는 3-dupack의 원인 판단하기 어려운 상태이므로 다음 패킷을 받을 때 더 명확한 판별을 하도록 현 상태를 유지한다.

## 3. 제안아이디어

### 3.1 문제정의

TCP Westwood에서 사용되는 ABE 알고리즘은 패킷 손실의 원인에 대한 판단 없이 패킷 손실이 발생할 때 마다 전송량을 감소하는 단점이 있다. 이 문제를 해결하기 위해 TCP WestwoodVT는 패킷 손실의 원인을 링크 에러와 네트워크 혼잡으로 구분하고, 각각의 경우에 따라 전송량을 조절한다. 그러나 링크의 에러율이 높아지면 단위 시간당 송신자가 수신하는 ack의 수가 줄기 때문에 ABE를 적게 측정하게 된다. 또한 네트워크의 대역폭이 증가함에도 불구하고, 트래픽의 양이 증가하게 되면 ABE를 통해 도출된 전송량의 신뢰도가 떨어지는 문제점이 있다. 최근에 네트워크 형태가 유선 네트워크는 기가급의 대역

폭을 제공하고 무선 네트워크는 802.11n과 같이 100 메가급의 대역폭을 제공하는 환경에서는 기존의 Westwood에서 사용하는 ABE를 도입하게 되면 커진 혼잡원도우 값을 제대로 처리하지 못하고 가용대역폭을 제대로 활용하지 못하는 문제점이 있다. 이런 문제점을 해결하기 위해 TCP VEST는 패킷 손실의 원인을 구별한 후 slowstart threshold 값을 참고한다. 즉, 가용대역폭을 동적으로 획득하기 패킷 손실의 원인에 따라 slowstart threshold 값의 변화를 살피고 전송량을 조절한다.

### 3.2 패킷 손실 원인 구별법

TCP VEST는 네트워크에서 발생한 패킷 손실의 원인을 WestwoodVT와 동일한 방법을 사용한다. 송신자와 수신자 사이의 네트워크 버퍼의 상태를 파악하는 기법은 TCP Vegas의 방법을 적용한다. TCP VEST는 RTT가 가장 작게 측정된 minRTT와 모든 전송한 패킷에 대해 수신한 ack를 사용해 측정된 RTT를 갖는다. 이 두 개의 변수를 이용해서 네트워크의 상태가 최적할 때 보낼 수 있는 비율인 Expected 변수와 현재 네트워크의 상태로 보낼 수 있는 전송률인 Actual 변수 2개를 갖는다.

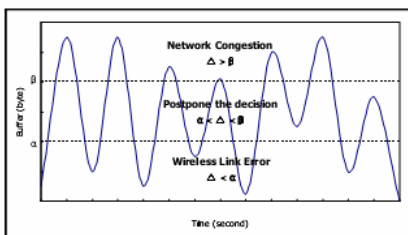
$$Expected = \frac{WindowSize}{minRTT} \quad \text{식 (1)}$$

$$Actual = \frac{WindowSize}{RTT} \quad \text{식 (2)}$$

Expected와 Actual 변수의 차이를  $\Delta$ 로 정의하고  $\Delta$ 는 현재 네트워크 노드의 버퍼에 존재하는 패킷의 수를 나타내며, 간접적으로 현재 네트워크의 상태를 정의할 수 있다.

$$\Delta = \left( \frac{WindowSize}{minRTT} - \frac{WindowSize}{RTT} \right) \times minRTT \quad \text{식 (3)}$$

추가적으로 TCP VEST에서는 네트워크 버퍼의 임계 수치를 최소 개수인  $\alpha$ 와 최대 개수인  $\beta$ 로 정의한다. 이 변수들은 네트워크 노드들의 상태를 반영하는데 쓰이며, TCP VEST에서 혼잡 회피모드(Congestion Avoidance)가 동작할 때 패킷 손실의 원인을 구분하는데 쓰이게 된다. 송신자가 3-dupack을 받게 되면 식 (1), (2), (3)을 이용하여  $\Delta$  값과  $\alpha, \beta$  값을 비교한다.



(그림 1) TCP VEST의 3-dupack 발생 시 원인 구별 방법

TCP VEST는 3-dupack을 받았을 때  $\Delta$  값이  $\alpha$ 보다 작은 경우에는 네트워크의 버퍼의 상태가 비어 있는데도 패킷이 손실되었으므로, 패킷 손실의 원인을 링크 에러로 가정한다. 그러나  $\Delta$  값이  $\beta$ 보다 큰 경우에는 네트워크 노드의 버퍼 상태가 차 있는 상태에서 패킷 손실이 발생했으므로, 손실 원인을 네트워크 혼잡으로 가정한다.  $\Delta$  값이  $\alpha$ 보다는 크고  $\beta$ 보다는 작을 경우에는 상황 파악이 어렵기 때문에 원인 구분을 연기하게 된다.

### 3.3 Effective ssthreshold Control(ESC)

고대역폭을 갖는 네트워크 안에서는 혼잡원도우 값을 조절하면, 송신자는 자신이 취할 수 있는 대역폭이 있음에도 불구하고 빠르게 증가하지 못하는 문제점이 발생하므로 갱신된 ssthreshold 값을 참조할 필요성이 있다. 따라서 패킷 손실이 발생했을 때, 손실 원인이 링크 에러 때문일 경우에는 송신자는 갱신된 ssthreshold 값을 따라서 자신이 네트워크에 혼잡을 끼치지 않는 범위 안에서 혼잡원도우 값을 최대로 증가시키는 방법을 제시한다. 만일 패킷 손실 원인이 네트워크 혼잡일 경우에는 무선 네트워크 환경에서 우수한 전송률을 보장하는 Westwood의 ABE를 이용하여 전송률을 재계산한다. 패킷 손실의 원인을 정확하게 판단할 수 없는 경우에는 패킷 손실 발생 이전의 ssthreshold 값(pre\_ssthreshold)과 발생 직후의 ssthreshold 값을 비교한다. 만일 pre\_ssthreshold 값이 ssthreshold 값보다 크다면, 현재의 네트워크 링크는 잔여 대역폭이 더 있음을 의미하므로, 네트워크의 상태가 좋아지고 있는 것으로 가정하고 송신자의 혼잡원도우 값을 1개 증가시킨다. 그러나, 만일 pre\_ssthreshold 값이 ssthreshold 값보다 작거나 같다면, 링크에 잔여대역폭 양이 줄어든 것을 의미하고, 현재 네트워크의 상태가 불안한 것이기 때문에 송신자가 가지고 있는 혼잡원도우 값을 그대로 유지한다.

```

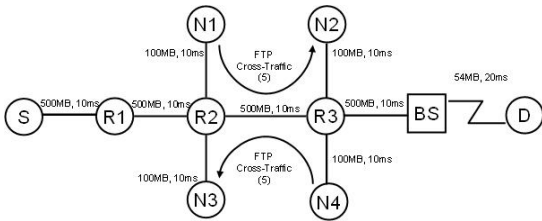
/* In Congestion Avoidance */
if (3-dupack are received)
    if ( $\Delta > \beta$ )
        /* if 3-dupack due to congestion */
        bandwidth estimation of westwood;
    else if ( $\Delta < \alpha$ )
        /* if 3-dupack due to bit error rate */
        ssthresh = BWE * minRTT / segment_size;
        cwnd = ssthresh;
    else
        /* comparing the ssthreshold value */
        pre_ssthresh = ssthresh;
        ssthresh = BWE * minRTT / segment_size;
        if (pre_ssthresh < ssthresh)
            cwnd = cwnd ++;
        end if
    end if
end if
    
```

(그림 2) TCP VEST의 혼잡제어 Pseudo Code

4. 성능평가

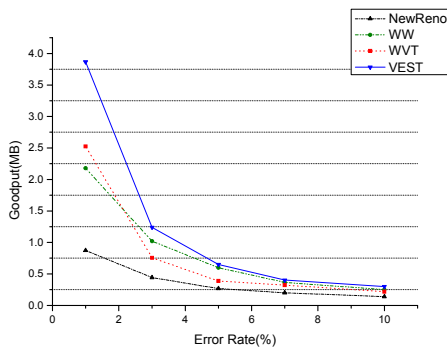
본 장에서는 NS-2[7] 시뮬레이터를 사용하여 TCP VEST의 성능 지표 중 goodput을 TCP New Reno, Westwood, WestwoodVT와 비교한다. 시뮬레이션에 사용된 네트워크는 무선 링크의 대역폭이 50MByte를 갖는 초고속 무선 네트워크 환경으로 구성하고 진행한다.

(그림 3)과 같이 구성한 초고속 무선 네트워크 안에서 동작하는 TCP VEST의 성능을 평가한다. 송신자에서 수신자 사이에는 두 개의 라우터가 존재하며 N1 - Router2 - Router3 - N2 방향으로 5개의 FTP 백그라운드 트래픽이, N4 - Router3 - Router2 - N3 방향으로 5개의 FTP 백그라운드 역방향 트래픽이 존재한다. 또한 Base Station과 수신자간의 대역폭은 50MBytes로, 딜레이는 20ms으로 설정한다.



(그림 3) 초고속 무선 네트워크 토폴로지

(그림 4)와 같이 TCP VEST는 다른 스킴들과 비교했을 때 초고속 네트워크 안에서도 우수한 성능을 나타낸다. TCP VEST는 WestwoodVT와 비교했을 때 20%에서 50%의 goodput의 향상을 보이고, 링크의 에러율이 1%인 경우에 WestwoodVT의 경우 2.5 Mbytes의 대역폭을 취그러나, TCP VEST는 4 MBytes의 대역폭을 취한다. 이를 통해 TCP VEST가 초고속 네트워크에서 동작할 때 네트워크의 가용대역폭을 원활하게 취하는 것을 알 수 있다.



(그림 4) 초고속 네트워크 안에서 TCP VEST의 goodput

5. 결론

본 논문에서는 송신자 기반 무선네트워크 환경에서 패킷 손실의 원인을 구분하는 WestwoodVT를 개선하여 초

고속 네트워크 환경에서도 높은 성능을 낼 수 있는 TCP VEST(Vegas-based Estimator with Slowstart Threshold)를 안한다. 본 기법에서는 패킷 손실이 발생했을 때 발생 원인을 링크의 에러와 네트워크 혼잡으로 구분하고, 우선적으로 패킷 손실이 발생할 때마다 ssthreshold 값을 재계산한다. 패킷 손실의 원인이 링크의 에러인 경우에는 송신자가 가지고 있는 전송윈도우 값을 갱신된 ssthreshold 값으로 설정하여, 자신의 송신자가 현재 네트워크의 혼잡을 유발하지 않고 최대로 보낼 수 있는 전송량을 반영해 높은 전송률을 보장한다. 또한, 패킷 손실의 원인이 네트워크 혼잡인 경우에는 일반 Westwood에서 사용하는 가용대역폭 측정 알고리즘을 사용하여 안정적인 전송률을 보장하였다. 패킷 손실의 원인을 정확하게 구별하지 못하는 경우에는 ssthreshold 값의 패킷 손실 발생 이전 이후의 값을 비교하여 네트워크 상태의 변화를 예측하고 적절하게 대응하도록 한다. NS-2를 통한 성능 평가에서 TCP VEST와 New Reno, Westwood, WestwoodVT의 처리량을 비교했을 때 제안 기법은 초고속 네트워크에서 동작할 때 모든 링크 에러율 범위 안에서 20%에서 50%까지의 향상된 전송률을 보장하고, 특히 가용대역폭 측정 관점에서 WestwoodVT보다 1.5배 이상 취할 수 있음을 보인다.

ACKNOWLEDGMENT

본 연구는 지식경제부 및 정보통신연구진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음 (IITA-2008-(C1090-0801-0046))  
교신저자 : 추현승

참고문헌

[1] J. Postel, "Transmission control protocol," RFC 793, 1981.  
 [2] M. Allman, V. Paxson and W. Stevens, "TCP Congestion control," RFC 2581, 1999.  
 [3] Y. Tian, K. Xu, and N. Ansari, "TCP in wireless environments: problems and solutions," IEEE Radio Communications, vol. 43, no. 3, pp.S27-S32, Mar 2005.  
 [4] D. Chiu, and R. Jain, "Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks," J.Comput. Networks, vol. 17, no. 1, Jun. 1989.  
 [5] J. Koo, S. Mun, H. Choo, "WestwoodVT: Sender-Based Transmission Window Control Mechanism for Discriminating the Cause of Packet Loss", Springer-Verlag Lecture Notes in Computer Science, vol. 4479, pp. 391-402, May 2007.  
 [6] V. Jacobson, "Congestion avoidance and control," ACM SIGCOMM 1988, vol. 18, pp. 314-329, Aug 1988.  
 [7] UCB/LBNL/VINT Network Simulator [Online]. Available: <http://www.isi.edu/nsnam/ns>