한국항행학회 논문지 제 10 권 제 1 호 2006년 3월

유무선 혼합 네트워크에서의 TCP Fairness 향상 기법

A TCP Fairness Improvement Scheme for Wired and Wireless Heterogeneous Networks

조성준*, 김성철*

Sung-Joon Cho* and Seong-Chul Kim*

요 약

유무선 혼합 네트워크에서 다수의 무선 단말기들은 한 접속점 (Access Point, AP)에 접속을 하여 통신을 한다. 이때 AP는 네트워크에서의 병목점이 되어 버퍼오버플로우가 자주 발생하며, TCP 성능 저하의 주된 원인이 된다. 본 논문에서는 유무선 혼합 네트워크에서 AP의 버퍼 오버플로우를 방지하고, TCP fairness를 향상시키기 위한 새로운 알고리즘을 제안한다. 새 알고리즘은 AP의 버퍼 사용량에 따라 각 송신측의 전송률을 적응적으로 제어하여 버퍼 오버플로우를 방지하고, 향상된 TCP fairness를 보장한다. 시뮬레이션을 통해 제안한 알고리즘을 사용했을 때, 버퍼 오버플로우 방지를 통해 처리율 및 TCP fairness가 향상됨을 검증하였다.

Abstract

Multiple wireless devices in wired and wireless heterogeneous networks communicate with counterparts via a single Access Point (AP). In this case, the AP becomes a bottleneck of the network, therefore buffer overflows occur frequently and result in TCP performance degradation. In this paper, the new algorithm that prevents buffer overflows at AP and enhances TCP fairness is proposed. Depending on the buffer usage of AP, the new algorithm adaptively controls each TCP senders' transmission rate, prevents buffer overflows and thus guarantees improved TCP fairness. It is proved that the algorithm makes better of TCP throughput and fairness by preventing buffer overflows.

Key words : TCP, fairness, heterogeneous networks, bottleneck

I. 서 론

무선 통신은 계속해서 광대역화, 고속화되고 있으며, 응용 서비스는 음성 통신 위주에서 데이터 통신 위주로 변화하고 있다. 데이터 통신 프로토콜로는 서비스의 종류에 따라 TCP, UDP, RTP/RTCP와 같은 다양한 전송 프로토콜이 사용되는데, 이 중 HTTP, FTP, Telnet과 같은 대부분의 상위 프로토콜들은 종단간 신뢰성이 보장되는 TCP 프로토콜을 전송 프로토콜로 사용하고 있다. 무선 환경에서도 유선과 마찬가지로 모바일 인스턴트 메신저, 네트워크 게임 등 신뢰성 높은 종단 대 종단 연결을 필요로 하는 응용 프로그램의 사용이 점차 증가함에 따라, TCP가 자주 쓰이고 있는데, TCP는 전송 에러율이 낮은 유선 환경을 기반으로 만들어졌기 때문에, 무선 환경과 같이 노이즈나 페이딩에 의한 전송 에

* 한국항공대학교 대학원 정보통신공학과 (Dept. of Inform. & Telecom. Eng., Graduate School of Hankuk Aviation University)
· 제1저자 (First Author) : 조성준
· 접수일자 : 2006년 1월 23일

러율이 높은 환경에서는 TCP 성능이 열화되는 문제가 발생한다 [1]. 이처럼 무선망에서의 TCP 성능 저하를 방지하기 위해 여러 기법이 개발되어 왔다 [2]-[4].

일반적으로 유무선이 혼합된 네트워크에서 다수의 통신 단말기들은 기지국과 같은 한 접속점(Access Point, AP)에 연결되어 통신을 한다. AP는 고속의 유선 구간과 상대적으로 저속인 무선 구간을 정합하는데, 이러한 대역폭 차이 때문에 많은 패킷들이 송수신될 때에는 한정된 버퍼 안에 계속해서 패킷이 쌓이게 된다. 따라서 네트워크에 부하가 많이 걸리는 경우 AP의 버퍼 오버플로우 (buffer overflow)에 의해 패킷 손실이 자주 일어나게 되는데, 패킷 손실은 TCP 송신측의 재전송 타임아웃(Retransmission Time Out, RTO)을 발생시켜 혼잡제어 알고리즘을 동작시키고 전송률을 낮추게 되므로 TCP 처리율(throughput)을 저하시키게 된다. 이처럼 유무선 혼합 네트워크에서 AP는 병목점(bottleneck)이 되어서 TCP 전송률 저하의 주된 원인이 된다.

AP의 버퍼 오버플로우에 의해 다량의 패킷이 손실될 때, 네트워크의 병목점인 AP에 연결된 모든 송신측들이 동시에 혼잡제어 알고리즘을 동작시켜 전송률을 낮추게 되는 것을 global synchronization 이라고 한다 [5]. 이때 각각의 송신측은 서로 다른 Round-Trip Time (RTT)을 갖게 되어 다시 전송률을 증가시키는 정도에 차이가 발생한다. 작은 RTT를 갖는 송신측은 전송률을 빨리 올릴 수 있는 반면, 큰 RTT를 갖는 송신측은 전송률을 천천히 올리게 된다. 이에 따라 시간이 지나면서 TCP flow별로 서로 다른 대역폭을 차지하게 되고, TCP fairness는 저하된다. 대역폭과 데이터 전송 속도와는 밀접한 관계에 있으므로 TCP fairness가 좋고 나쁨에 따라 동일한 AP를 통해 통신을 하는 여러 사용자들이 비슷한 전송 속도로 혹은 서로 상이한 전송속도로 통신을 하게 된다.

TCP fairness는 사업자 측면에서는 중요한 고려사항이 되는데, 정액제 요금체계를 채택한 경우 TCP fairness가 나쁘게 되면 사용자들에게 불공평한 서비스를 제공하게 되기 때문이다.

본 논문에서는 유무선 혼합 네트워크에서 AP의 버퍼 오버플로우를 방지하면서 TCP fairness를 향상시키기 위한 새로운 알고리즘을 제안한다. 새 알고리즘은 AP의 버퍼 사용량에 따라 각 TCP 송신측의 전송률을 적응적으로 제어하여 버퍼 오버플로우를 방지하고, TCP fairness를 향상시킨다. 시뮬레이션을 통해, 제안한 알고리즘을 적용했을 때 버퍼 오버플로우를 상당부분 방지하고, AP의 버퍼가 많이 채워졌을 시에 각 TCP flow별 전송률을 기존의 송수신한 패킷 양에 따라 차등 조정함으로써 TCP fairness가 향상됨을 검증한다.

본 논문의 구성은 다음과 같다. Ⅱ장에서 버퍼 오버플로우 방지와 TCP fairness를 향상시키기 위한 알고리즘을 제안·설명한다. Ⅲ장에서는 시뮬레이션 모델과 시뮬레이션 결과를 분석하고, 마지막으로 Ⅳ장에서 결론을 내린다.

Ⅱ. 버퍼 오버플로우 방지 및 TCP Fairness 향상 알고리즘

본 논문에서 제안하는 AP 버퍼 오버플로우 방지 및 fairness 향상 알고리즘은 AP의 버퍼 사용량에 따라 각각의 TCP flow마다 송수신되는 패킷의 양을 조정하도록 되어있다.

AP의 버퍼 오버플로우를 방지하려면 송신측의 패킷 전송률을 조절할 수 있어야 하는데, 제안된 알고리즘에서는 TCP 헤더에 포함되어 전송되는 advertisement window (awnd)를 이용한다. awnd는 수신측이 송신측에게 수신 가능한 패킷 수를 알려줌으로써, 수신측까지 도착한 패킷이 버려지지 않도록 하는 기능을 하며, TCP 패킷을 전송하는 송신측은 이 awnd와 네트워크의 혼잡을 피하기 위해 사용하는 congestion window (cwnd)를 비교해 작은 값을 선택하여 전송률을 결정하는데, 이것은 이 전송률보다 높은 전송률로 패킷 전송을 할 경우 네트워크에 혼잡이 발생 하거나, 수신측에서 수신을 하지 못할 수 있기 때문이다. cwnd는 송신측 내부에서 계산되는 값이지만, awnd는 TCP 헤더에 포함되어 전송되므로, 패킷 전송을 중계하는 AP에서 버퍼 사용량에 따라 awnd 값을 적절히 변경하여 송신측의

전송률을 조절하고 버퍼 오버플로우를 방지할 수 있다.

TCP fairness 향상을 위해서는 각 flow별 전송률을 차등적으로 조절하여야 한다. 즉, AP의 버퍼 사용량이 높아져 송신측의 전송률을 낮춰야 할 때 많은 양의 패킷을 전송하고 있던 송신측은 전송률을 많이 낮추고, 적은 양의 패킷을 전송하고 있던 송신측은 전송률을 조금 낮추게 해야 한다. 이로써 TCP flow별로 비슷한 수준의 버퍼 공간 사용과 대역폭 점유율을 갖게 함으로써 TCP fairness를 향상시킬 수 있다.

AP에서 awnd 값을 설정하여 TCP 송신측의 전송률을 제어하는 알고리즘의 순서도는 다음과 같다.

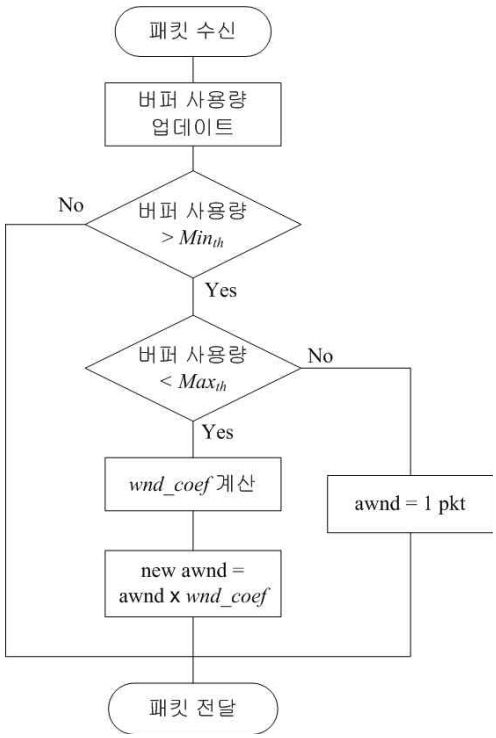

그림 1. 제안하는 알고리즘의 순서도
Fig. 1. A flowchart of the proposed algorithm.

그림 1에서 Max_{th}, Min_{th}, *wnd_coef*는 제안하는 알고리즘에 쓰이는 변수로서, Max_{th}와 Min_{th}는 알고리즘이 동작하는 방법을 결정하는 미리 설정된 threshold 값이다. *wnd_coef*는 알고리즘 동작 시 송신측의 전송률을 조절하는데 쓰이는 변수이다. 그림 2와 같이 결정되는 *wnd_coef*는 1과 같거나 작은 값을 갖는데, 이 값과 AP를 통해 전송되는 패킷의 awnd 값을 곱하여 새로운 awnd 값을 계산하므로, 새로 계산된 awnd 값은 원래의 값보다 같거나 작게 된다.

또한, 이렇게 *wnd_coef*를 곱함으로써 원래 awnd 값이 크면 클수록 새로 계산된 awnd 값은 더 크게 감소한 값을 갖기 때문에, 많은 패킷을 송수신 하던 TCP flow일수록 송신측에서의 TCP 전송률이 많이 작아진다. 이로써, 결과적으로 AP에서의 버퍼 오버플로우 발생 빈도를 줄일 수 있고, TCP fairness를 향상시킬 수 있게 된다.

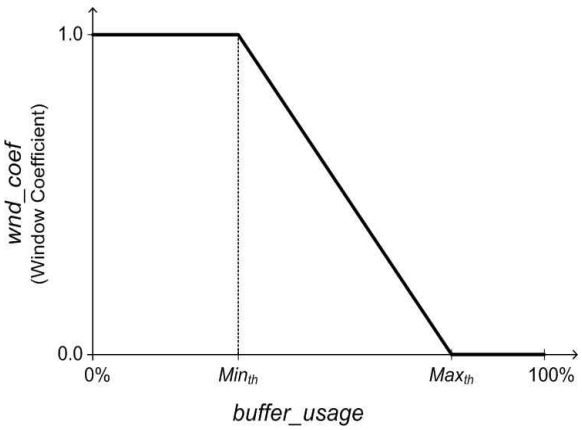

그림 2. Window coefficient.
Fig. 2. Window coefficient.

알고리즘의 구체적인 동작과정은 다음과 같다.

1) AP는 매번 패킷을 수신할 때마다 자신의 버퍼 사용량 *(buffer_usage)*을 수시로 모니터링 한다.

2) 버퍼 사용량이 일정 수준 (Min_{th}) 이상 올라가서 알고리즘이 동작할 수준이 되었는지 판단한다. 버퍼 사용량이 Min_{th}보다 낮을 때는 전송률을 조절할 필요가 없으므로 아무 작업도 수행하지 않고 패킷을 수신측으로 전달한다.

3) 버퍼 사용량이 Min_{th}보다 높을 때는 전송률을 조절하기 위해 새로운 awnd 값을 계산한다. 이 때 버퍼 사용량이 Max_{th}보다 낮은 경우 그림 2와 같이 *wnd_coef*를 계산하고, 이 값을 원래의 awnd 값에 곱하여 새로운 awnd 값을 얻은 후, 전달할 패킷의 TCP 헤더에 기록한다.

4) 버퍼 사용량이 Max_{th}보다 높아서 심각한 버퍼 오버플로우가 임박한 경우, 버퍼 공간에 다시 여유

가 생길 때까지 패킷 전송을 최대한 억제하기 위해 awnd 값을 1로 설정한다. 따라서 패킷 전송을 하던 모든 TCP flow의 전송률이 최소한으로 유지되고, 버퍼 오버플로우를 방지한다.

위에서 제안한 알고리즘에 의해 각 TCP 송신측에서는 AP의 버퍼 사용량에 따라 적응적으로 패킷 전송률을 조절하게 되므로 AP의 버퍼 오버플로우를 방지하고 TCP fairness를 향상시킬 수 있게 된다.

Ⅲ. 시뮬레이션

제안하는 알고리즘의 성능평가를 위하여 버클리 소재 캘리포니아 주립대학의 ns-2를 사용하였다[6].

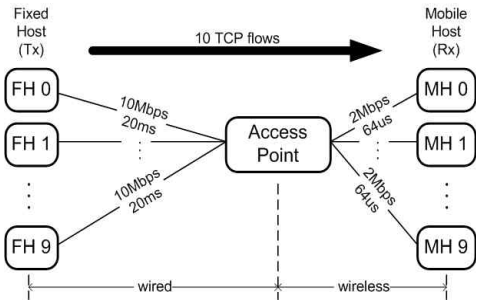

그림 3. 시뮬레이션 모델
Fig. 3. The simulation model.

시뮬레이션 모델은 그림 3과 같이 10개의 TCP 송신측 (Fixed Host, FH)과 10개의 TCP 수신측 (Mobile Host, MH)이 하나의 AP를 거쳐 데이터를 주고받는 환경을 가정하였다. 각각의 송신측과 수신단이 하나의 flow를 형성하며 총 10개의 TCP flow가 존재한다. TCP 세그먼트는 FH에서 MH로만 전송되고, 송신측인 FH에서는 FTP traffic이 계속 생성된다고 가정하였다. 유선 링크는 10Mbps의 대역폭과 20ms의 전송지연을 가지는 이더넷 (Ethernet) 망이며, 무선링크는 2Mbps의 대역폭과 64μs의 전송지연을 가지는 2.3GHz 휴대인터넷망을 가정하였다. 전송계층 프로토콜은 TCP-Reno를, AP에서의 버퍼는 DropTail 버퍼를 사용하였다.

그리고 무선 링크에서의 패킷 손실은 상향 링크를 통해 전송되는 패킷 (ACK 패킷)보다 상대적으로 훨씬 긴 길이의 패킷 (TCP 세그먼트)이 전송되는 하향 링크에서만 발생한다고 가정하였고, 무선 링크의 에러모델은 two-state Markov 모델을 사용하였다[7]. 그 밖의 시뮬레이션 파라미터는 표 1과 같다.

최고의 TCP fairness를 위한 Max. threshold (Max_{th}), Min. threshold (Min_{th}) 값들은 네트워크의 대역폭이나 지연시간, 그리고 AP의 버퍼 크기 등 다양한 변수에 따라서 각각 다르게 된다. 따라서 시뮬레이션은 임의의 threshold 값이 제안한 알고리즘의 성능에 미치는 영향을 알아보기 위해, Max_{th}와 Min_{th} 값을 바꾸면서 수행하였으며, threshold 값을 바꿀 때마다 random seed를 변경하면서 10번씩 수행한 평균을 결과로 사용하였다.

표 1. 시뮬레이션 파라미터
Table 1. Simulation parameters.

Parameter	Value
Number of flows	10
Transport Protocol	TCP-Reno
Application Data Type	FTP
Maximum Segment Size	1040 bytes
AP Buffer Type	DropTail
AP Buffer Size	30 packets
Max. threshold (Max_{th})	50%~90% of *buffer_size*
Min. threshold (Min_{th})	20%~80% of *buffer_size*
Error Model	Two-state Markov
Wired Link PLR*	0
Wireless Link PLR*	10^{-2}

*PLR, packet loss rate

우선 TCP fairness 향상을 위한 아무런 알고리즘도 사용하지 않았을 때의 시뮬레이션 결과 중 한 예를 그림 4에 나타낸다.

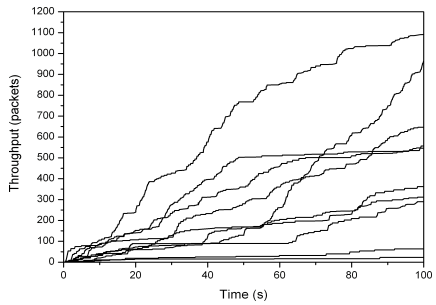

그림 4. TCP만 쓰였을 때 각 flow 별 처리율
Fig. 4. Each flow's throughput when only plain TCP was used.

시뮬레이션 결과 그래프에서 가로축은 시간을, 세로축은 전송된 패킷의 TCP sequence number, 즉 TCP 처리율 (throughput) 을 나타낸다. 그림 4에서 보이는 바와 같이, 총 10개의 TCP flow 중 어떤 flow는 높은 처리율을 보이는 반면, 어떤 flow는 매우 낮은 처리율을 보이고 있다. 모든 flow에서 동일한 application (FTP)이 사용되도록 했으므로 비슷한 양의 패킷 전송이 이루어짐에도 불구하고, 어떤 flow는 충분히 많은 양의 패킷을 보내고, 어떤 flow는 패킷을 거의 보내지 못함으로써, 전송률의 편차는 약 1073 패킷 (1096 패킷 대 23 패킷)으로 매우 큰 차이가 나타남을 확인할 수 있다. AP에서 버퍼 오버플로우가 발생하는 경우 global synchronization 현상이 발생하는데, 무선 구간에서의 패킷 손실 경향 및 그에 따른 backoff 시간의 차이로 인하여 TCP 송신측에서의 전송률 증가 속도가 달라지기 때문이다.

다음으로 본 논문에서 제안한 알고리즘을 사용했을 때의 시뮬레이션 결과를 그림 5에 나타낸다.

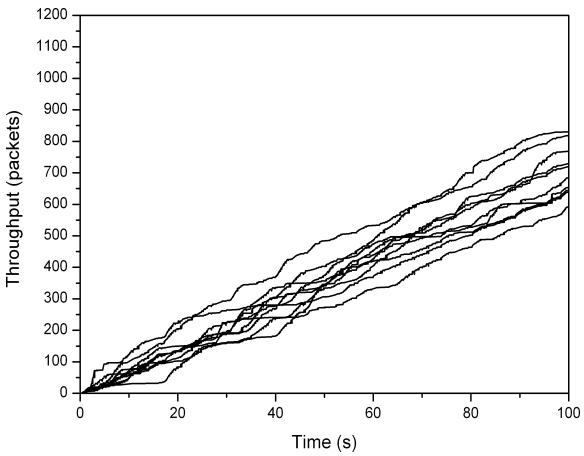

그림 5. 제안한 알고리즘이 쓰였을 때 각 flow 별 처리율 (Minth = 30%, Maxth = 80%)

Fig. 5. Each flow's throughput when the proposed algorithm was used. (Min_{th} = 30%, Max_{th} = 80%).

그림 5의 결과 그래프를 보면 총 10개의 TCP flow가 서로 비슷한 수준의 처리율을 보이고 있으며 처리율이 증가하는 양상 또한 서로 비슷함을 볼 수 있다. TCP fairness가 향상된 것은 AP에서 버퍼의 오퍼플로우를 방지하기 위해 어느 이상 버퍼가 사용되었을 때 각 TCP 송신측의 전송률을 차등적으로 낮추기 때문이다. 즉, 많은 수의 패킷을 전송하며 대역폭을 과도하게 차지하던 송신측은 전송률을 많이 낮추고, 적은 수의 패킷을 전송하며 매우 적은 대역폭만을 차지하던 송신측은 전송률을 조금 낮추게 함으로써, 결국 TCP flow별로 비슷한 수준의 버퍼 공간 사용과 대역폭 점유율을 갖게 되었다.

그림 4와 그림 5의 결과를 정량적으로 비교하기 위해 식 1로 표현되는 fairness index [8]를 계산하여 사용한다.

$$\text{Fairness index} \equiv \frac{\left(\sum_{i=1}^{n} b_i\right)^2}{n \times \left(\sum_{i=1}^{n} b_i^2\right)} \qquad (1)$$

여기서 n은 총 TCP flow의 수를 나타내며, b_i는 i 번째 flow의 처리율을 나타낸다. Fairness index는 0부터 1까지의 값을 갖는데, fairness가 좋을수록 1에 가까운 값을, fairness가 나쁠수록 0에 가까운 값을 갖는다.

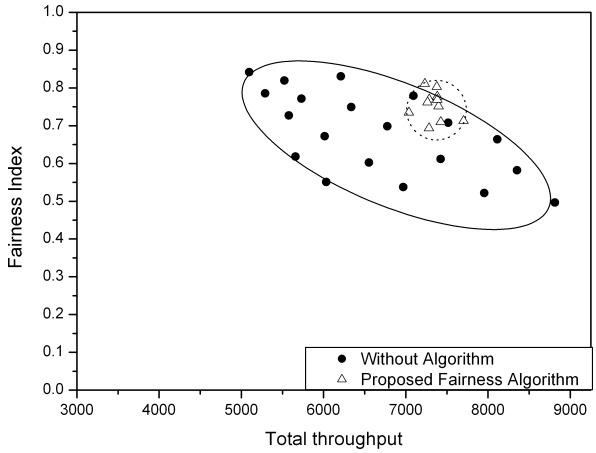

그림 6. 제안한 알고리즘이 쓰인 경우와 쓰이지 않은 경우의 전체 처리율과 fairness의 관계

Fig. 6. The relationship between total TCP throughput and TCP fairness whether the proposed algorithm was used or not.

그림 6은 전체 처리율 (total throughput)과 fairness index와의 관계를 나타내는 그래프로서, 전체 처리율이란 모든 TCP flow들의 처리율의 합을 의미한다. 각 점은 한 번의 시뮬레이션 결과를 나타

내고, 제안한 알고리즘을 사용했을 때와 사용하지 않았을 때를 각각 구별하여 표시하였다. 그림 6의 그래프로부터, 제안한 알고리즘이 사용되지 않았을 때는 처리율이 높아질수록 fairness가 나빠지고, fairness가 좋아질수록 처리율이 떨어지는 것을 확인할 수 있다. 즉 TCP 처리율과 TCP fairness가 trade-off 관계에 있음을 알 수 있다. 하지만 제안한 알고리즘이 쓰였을 때는 항상 비슷한 수준의 처리율과 TCP fairness를 보여 모든 송신측이 원활한 패킷 전송을 할 수 있다.

제안한 알고리즘에 의한 결과에서도 어느 정도의 편차가 나타났는데, 이것은 시뮬레이션 경우마다 결과가 다르기 때문에 발생하게 된다. 이 편차의 정도는 알고리즘에서 사용하는 Max_{th}와 Min_{th} 값의 설정과 밀접한 관계에 있는데, 적절치 못한 threshold 값을 사용했을 경우 시뮬레이션 결과마다 나타나는 편차가 크고, 적절한 threshold 값을 사용했을 경우 편차가 줄어들었다. 최적의 threshold 값을 설정하는 것은 시뮬레이션 환경과 트래픽의 종류에 따라 다른 것으로 나타났으며, 보다 정확하게 최적의 threshold 값을 결정하는 방법은 추후 연구가 필요하다.

제안한 알고리즘을 사용하는 경우, AP에서 버퍼 오버플로우가 발생하는 경우를 많이 줄여 TCP 처리율의 향상을 가져올 것으로 예상했으나, 실제로는 알고리즘을 적용하지 않았을 경우와 비교해 큰 차이는 없었다 (미적용시 전체 처리율 : 평균 6852 패킷, 적용시 전체 처리율 : 평균 7243 패킷으로 차가 오차범위 수준임). 이는 제안한 알고리즘이 AP의 버퍼 오버플로우를 미연에 방지하기 위해, 일정 수준 이상으로 버퍼가 사용되고 있을 때 TCP 송신측의 전송률을 제한하기 때문이다. 즉, AP의 버퍼 공간에 어느 정도의 여유가 있고, TCP 송신측의 전송 윈도우 (congestion window)가 커서 더 많은 데이터를 송신할 수 있음에도 불구하고, AP의 버퍼 오버플로우로 인한 global synchronization을 방지하기 위해 송신측의 전송률을 제한하기 때문이다.

Ⅳ. 결 론

본 논문에서는 유무선 혼합 네트워크에서 AP의 버퍼 오버플로우를 방지하고 TCP fairness를 향상시키는 알고리즘을 제안하고 시뮬레이션을 통해 그 성능을 검증하였다.

유무선 혼합 네트워크에서 기존의 TCP만으로는 AP의 버퍼 오버플로우에 의한 패킷 손실이 처리율 저하 및 TCP fairness 악화를 초래한다. 본 논문에서 제안한 알고리즘은 AP의 버퍼 사용량을 감시하며, 그 사용량이 증가해 버퍼 오버플로우가 발생되기 전에 송신측의 패킷 전송률을 낮추도록 한다. 이때 전송률이 높은 송신측은 전송률을 많이 낮추도록 하고, 전송률이 낮은 송신측은 전송률을 조금만 낮추도록 하여, 버퍼 오버플로우를 방지하고 TCP fairness 또한 향상 시킨다. 시뮬레이션 결과, 아무런 알고리즘도 사용되지 않았을 때 모든 송신측들이 같은 환경에서 패킷 전송을 함에도 불구하고, 각각의 처리율의 편차가 매우 컸다. 즉, 어떤 송신측은 많은 패킷을 보내 높은 처리율을 얻을 수 있었지만, 어떤 송신측은 극히 소수의 패킷만을 보낼 수 있었다.

하지만 제안한 알고리즘을 적용한 시뮬레이션에서는 모든 송신측들의 처리율간 편차가 거의 없었다. 이는 AP의 버퍼 오버플로우를 방지하기 위해 송신측의 전송률을 조절할 때 차등적으로 조절하기 때문이다. 패킷 전송을 많이 하던 송신측의 전송률은 많이 낮추고, 패킷 전송을 많이 하지 않던 송신측의 전송률은 조금만 낮춤으로써, TCP fairness를 향상시킬 수 있었다.

한편 시뮬레이션을 통해 살펴본 결과에 의하면 TCP 처리율과 TCP fairness는 trade-off 관계에 있어서, 기존의 TCP만 사용된 경우 처리율이 높을 때는 fairness가 좋지 않고, fairness가 좋을 때는 처리율이 낮아지는 관계가 나타났다. 다수의 시뮬레이션 결과에서 확인할 수 있듯이 처리율과 TCP fairness의 편차가 커서 안정적인 TCP 성능을 보장하지 못한다 [9]. 따라서 유무선 혼합 네트워크에서 TCP 처리율뿐만 아니라 TCP fairness 또한 TCP 성능의 중요한 척도이며, TCP fairness를 향상시키기 위한 알고리즘이 필요함을 알 수 있다.

본 논문에서는 유무선 혼합 네트워크에서 TCP

처리율과 TCP fairness를 동시에 악화시키는 원인인 AP의 버퍼 오버플로우를 방지하기 위한 알고리즘을 제안하였다. 제안한 TCP fairness 향상 알고리즘을 사용하는 경우, AP의 버퍼 사용량에 따라 적응적으로 송신측의 전송률을 제어하여 AP의 버퍼 오버플로우를 방지함으로써, 버퍼 오버플로우에 의한 패킷 손실이 없어져 각각의 송신측이 혼잡제어를 시작하는 횟수를 줄일 수 있고, 다수의 시뮬레이션 결과에서 꾸준히 높은 수준의 처리율을 보였다. 또한 여러 TCP flow들이 서로 비슷한 대역폭을 차지하도록 하여 TCP fairness를 향상시킬 수 있었다.

본 논문의 연구 결과는 조만간 상용화 될 WiBro나 Mobile WiMAX 같은 무선 광대역 통신망을 설계 시 반영되어 보다 효율적인 통신 시스템 구축에 활용될 수 있을 것이다.

감사의 글

이 논문은 2005년도 한국항공대학교 교비지원 연구비에 의하여 지원된 연구의 결과임.

참 고 문 헌

[1] 원기섭, “모바일 네트워크에서 TCP 처리율 향상을 위한 Snoop+α 프로토콜,” 한국항공대학교 석사학위논문, Feb. 2005.

[2] H. Balakrishnan, S. Seshan and R.H. Katz, "Improving reliable transport and handoff performance in cellular wireless networks," *ACM Wireless Net works*, vol. 1, no. 4, pp. 469-481, Dec. 1995.

[3] H. Balakrishnan *et al.*, "A comparison of mechanisms for improving TCP performance over wireless links," *IEEE/ACM Trans. Networking* (*TON*), vol. 5, no. 6, pp. 756-769, Dec. 1997.

[4] E. Ayanoglu *et al.*, "A Link-Layer Protocol for Wireless Networks," *ACM/Baltzer Wireless Networks J.*, vol. 1, pp. 47-60, Feb. 1995.

[5] V. Jacobson, "Congestion avoidance and control," *Symp. proc. Comm. architectures and protocols* (*SIGCOMM'88*), vol. 18, no. 4, pp. 314-329, Aug. 1988.

[6] VINT Project, "The Network Simulator - ns-2,"; http://www.isi.edu/nsnam/ns.

[7] M. Zorzi, R.R. Rao and L.B. Milstein, "On the accuracy of a first-order Markov model for data transmission on fading channels," *Int'l Conf. Universal Personal Comm.* (*ICUPC*), vol. 6, no. 10, pp. 211-215, Nov. 1995.

[8] F. Peng and V. C. M. Leung, "Enhancing fairness and throughput of TCP in heterogeneous wireless networks," *Personal, Indoor and Mobile Radio Comm., PIMRC 2003. 14th IEEE Proc.*, vol. 1, no. 1, pp. 94-98, Sept. 2003.

[9] 김성철, 조용범, 문일영, 조성준, “유무선 혼합망에서 TCP의 공평한 대역폭 분배를 위한 연구,” *한국해양정보통신학회 춘계학술대회*, vol. 9, no. 1, pp. 113-116, May 2005.

조 성 준 (趙成俊)

1969년 2월 : 한국항공대학교 항공통신공학과(공학사)
1975년 2월 : 한양대학교 대학원 (공학석사)
1981년 3월 : 오사카대학 대학원 (공학박사)
1972년 8월~현재 : 한국항공대학교 항공전자 및 정보통신공학부 교수
관심분야 : 이동통신, 무선통신, 환경전자공학, 이동무선 인터넷

김 성 철 (金成哲)

2005년 2월 : 한국항공대학교 정보통신공학과(공학사)
2006년 2월 : 한국항공대학교 대학원 정보통신공학과 석사과정
관심분야 : Wireless TCP, TCP Fairness