

비대칭 링크에서 TCP segment 전송을 위한 수신단에서의 버퍼 오버플로어 방지

강제완, 장주욱
서강대학교 전자공학과

Prevention of buffer overflow for TCP segment transmission in asymmetric link

Jea-Wan Kang, Ju-Wook Jang
Dept. of Electronic Engineering, Sogang Univ.

요 약

비대칭 링크 상에서 TCP 전송을 향상 위한 새로운 기법을 제시하였다. 기존 TCP는 비대칭 링크에서 수신단의 전송버퍼의 오버플로우에 인하여 패킷의 재전송, 타임아웃 등으로 인하여 전송율이 상당히 낮아지는 문제점을 가진다. 본 논문에서는 이러한 단점을 극복하기 위해 수신단의 전송 버퍼가 가득 찰 경우, 이를 송신단에게 통보하여 송신단은 전송을 중단하고, 혼잡 윈도우를 유지함으로써, 전송 버퍼의 여유가 생겨서, 수신단이 송신단에 이를 통보하여 다시 전송이 가능해졌을 경우, 송신단이 즉시 전송을 재개할 수 있도록 하였다. 제안된 기법의 성능 향상을 시뮬레이션을 통해 확인하였다.

1. 서 론

비대칭 링크에서 기존의 TCP는 비대칭성에 의해 수신단의 전송버퍼의 오버 플로우가 발생되어 ACK 패킷 손실이 되고, 이로 인하여 송신단이 패킷 재전송, timeout 발생에 의한 전체 전송율이 상당히 낮아지는 문제점을 발생하게 되었다.

본 논문에서는 비대칭 링크에서 발생되어지는 문제점을 살펴보고, 이를 바탕으로 한 기존 기법보다 TCP 전송율을 향상시킬 수 있는 기법을 제안한다.

2. TCP 혼잡 제어 기법

TCP는 측정된 네트워크 혼잡 정도에 따라 전송하는 데이터의 양을 조절하는 혼잡제어 기법을 몇가지를 가지고 있다. 이러한 기법은 네트워크의 안정성을 제공하고, 각 연결이 가능한 최고의 효율을 갖도록 한다. 대표적인 혼잡 제어 기법인 Slow Start, 혼잡 회피 기법에 대해 살펴보겠다(1)(2).

2-1. Slow Start

Slow Start 기법은 네트워크에 전송되는 데이터 양을 점진적으로 증가시키기 위한 기법이다. 네트워크 상황에 적절한 전송 가능 데이터량을 제한하는데 이를 혼잡 윈도우(Congestion window: 이후 cwnd로 표기)라 한다. 전송을 시작할 때 윈도우의 크기를 1 세그먼트로 초기화한다. 따라서 송신단은 1세그먼트로 초기화하여 이를 전송한다. 데이터 전송후 수신된 ACK에 대해서 혼잡 윈도우는 1세그먼트씩 증가하게 되고 따라서 혼잡 윈도우의 크기는 RTT 마다 지수적으로 증가하게 된다. 이때 데이터 전송 속도가 너무 급격히 증가해서 패킷 손실이 일어 날 수 있는데 이러한 문제점을 해결하기 위해 혼잡 회피 기법이 쓰인다.

2-2 혼잡 회피 기법

Slow Start 기법과 달리, ACK를 받을 때마다 혼잡 윈도우의 크기를 $1/cwnd$ 만큼 증가시킨다. 따라서 수신단이 매 세그먼트마다 ACK를 보낼 때 혼잡 윈도우는 매 RTT마다 1세그먼트씩 증가하게 되는 것이다(3).

3. 비대칭 링크와 다중 분할다운 방식

3-1 비대칭 링크

인터넷 사용자의 증가와 멀티미디어의 전송으로 인하여 개인 사용자에게 더욱 더 빠른 데이터 전송이 필요하게 되었고, 이에 따라 널리 사용하는 전송 매체가 ADSL과 Cable Modem이다. 이 방식은 인터넷 서비스의 특성상 비대칭 링크를 사용한다. ADSL의 경우 이론상 대역폭이 2.7Km에서 다운로드 6.1Mb/s, 업 링크 640kb/s로 비대칭 성이 매우 크다

3-2 다중 분할 다운 방식

다중 분할 방식이란 대용량 데이터를 저용량으로 분할하여, 저용량의 데이터로 다운로드를 받아 대용량 데이터를 다운로드 받을 때 속도가 약간씩 줄어드는 현상을 최소화한 것이다. 본 논문에서는 비대칭 링크 상에서 이러한 방식의 사용시 문제점과 개선안을 제안할 것이다. 그림 1은 이러한 다중 분할 방식에 대해 간단히 나타내고 있다.

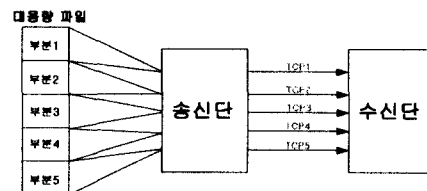


그림 1 다중 분할다운방식

4. 비대칭 링크에서 TCP의 문제점

대칭 링크 상에서 패킷 손실이 일어나는 주된 원인은 수신단의 전송링크 상에서 혼잡이 일어나는 것이다. 수신단의 전송링크 상에서 혼잡이 발생하여 ACK가 손실된다면 수신단이 패킷을 받았음에도 불구하고, 송신단에서는 timeout이 발생되고 Slow Start를 다시 시작하므로 전송을 감소를 초래한다.

5. 제안하는 TCP 전송을 향상 기법

비대칭 링크상에서 TCP segment 전송 시, 링크의 비대칭성의 해 수신 단의 전송버퍼에는 ACK를 보내기 위한 패킷들이 계속 쌓이게 되고, 이를 알지 못하는 송신 단은 계속해서 패킷을 전송한다. 결과적으로 비대칭 링크에서 수신단의 전송버퍼에는 오버플로우 현

상이 발생하여 전송 버퍼에서 drop된 ACK에 의해 송신단에선 이를 전송되지 않았다고 판단하고 이를 재전송하여 전체 전송율이 감소된다. 이에 따라 timeout이 발생되어 수신단의 전송버퍼의 오버플로우가 해결된다 하여도 송신단은 즉시 전송을 재개할 수 없고, 긴 timeout 시간 동안 기다려야 한다. 또한 timeout이 지나게 되고 cwnd가 1로 초기화되는 Slow Start를 수행하므로 상당한 전송율 감소로 이어진다. 이러한 문제점을 해결하기 위해 본 논문에서는 그림 2에서 TCP 헤더(4)에서 미래의 사용을 위해 예약된 6비트 필드(진한부분) 중 하나의 비트를 버퍼 상태를 나타내는 비트로 할당하여 수신단의 전송버퍼가 가득 찼을 경우는 이 비트를 1로 셋팅하여 송신단에 ACK 패킷을 전송함으로써 버퍼 상태를 통보한다. 이때 ACK 번호는 마지막으로 수신한 ACK 패킷 번호이다.

0		15 16		31	
16비트 송신지 포트 번호			16비트 수신지 포트 번호		
32비트 순서 번호					
32비트 ACK 번호					
4비트 헤더 길이	6비트 예약	Code bits		16비트 윈도우 크기	
		U R C K	A F S Y N	S S I N	
16비트 TCP checksum			16비트 긴급 포인터		
옵션					
데이터					

그림 2 FUL bit를 할당한 TCP header

수신단의 전송 버퍼가 가득 찬 상태를 통보받은 송신단은 전송을 중단하고, 혼잡 윈도우 크기를 유지하며, 재전송 timer를 동작시키지 않는다.

이후 ACK 패킷이 전송되어 수신단의 전송 버퍼의 빈 공간이 확보될 경우, 수신단의 FUL 비트를 0으로 셋팅하고 awnd(Advertisement Window, 이후 awnd로 표기)크기를 수신단의 버퍼의 빈 공간으로 하여 ACK 패킷을 송신단으로 전송한다. 송신단은 이 ACK 패킷 다음부터 전송을 재개하여 전송을 저하를 방지하였다. 이 과정을 그림으로 살펴보면 그림 3, 4, 5와 같다.



그림 3 수신단의 전송버퍼가 가득찬 경우

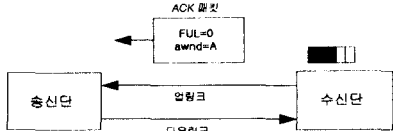


그림 4 수신단의 전송 버퍼의 빈 공간이 확보될 경우(A:사용가능한 공간)

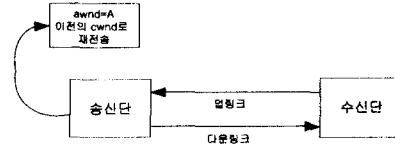


그림 5 송신단에서 전송을 재개할 경우

6. 실험 및 결과

본 논문에서 제안한 기법의 성능을 확인하기 위하여 시뮬레이션을 통해 구현한 후 실험하였다. 시뮬레이션 도구로는 TCP/IP 시뮬레이터인 버클리 대학의 ns-2 시뮬레이터를 사용하였다(5).

6.1 단일 TCP 흐름에 대한 성능 비교

그림 6은 기존 TCP와의 성능 비교를 위한 시뮬레이션 환경이다

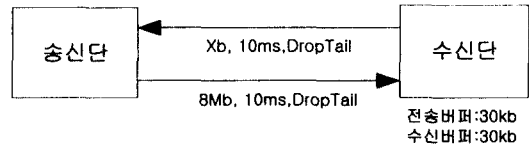


그림 6 성능비교를 위한 시뮬레이션 환경

업 링크의 속도를 달리하여 기존의 TCP와 제안된 TCP를 사용하여 비교하여 보았다. 여기서 사용된 TCP 버전은 TCP Tahoe이고 시뮬레이션 시간은 3.5초로 정하고 시뮬레이션을 해보았다. 그림 7는 업 링크가 각각 128kb/s, 64kb/s일 때, 기존의 TCP와 제안된 TCP의 성능을 나타내고 있다.

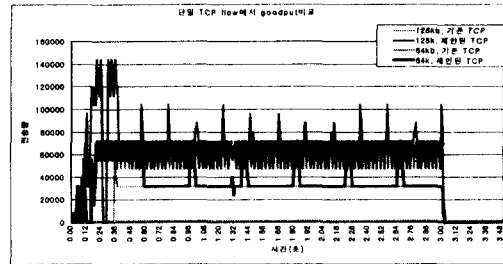


그림 7 단일 TCP flow에서 성능 비교

기존 TCP와 비교한 결과 업 링크가 128kb/s일 경우, 기존의 TCP flow에서는 goodput이 1828.57kb/sec이고, 제안된 TCP flow에서는 1952kb/sec이고, 업 링크가 64kb/s일 경우에는 기존의 TCP에서는 944kb/s이고, 제안된 TCP에서는 1069.71kb/s이다. 각각 6.75%, 13.31%의 전송률 향상됨을 알 수 있다.

6-2 비대칭 링크에서 다중 분할방식 성능 비교

제안된 TCP의 성능을 비교하기 위해 현재 널리 사용되고 있는 다중 분할다운 방식을 비대칭 링크 상에 사용 시 기존 TCP와 제안된 TCP의 성능을 비교하였다. 본 논문에서는 각 송신단과 수신단의 TCP flow를 5로 셋팅하여 시뮬레이션 하였다. 다중 분할다운 방식을 사용할 경우 전송 시작 시 다수의 TCP flow에 의해 빠른 전송율을 나타내지만, 비대칭 링크의 경우에는 수신단의 전송버퍼가 짧은 시간에 가득 차게 되어 전송율의 저하를 나타낼 수 있다. 그림 8에서 이러한 문제점을 보여주고 있다.

수신단의 전송버퍼에는 오버플로우 현상이 발생하며 전송버퍼에

서 drop된 ack에 의해 송신단에서는 이를 재전송하기 때문에 전체 전송율이 감소된다. 이에 따라 timeout이 발생되어 수신단의 전송 버퍼의 오버플로우가 해결된다 하여도 송신단은 즉시 전송을 재개할 수 없고, 긴 timeout 시간 동안 기다려야 한다. 또한 timeout이 지나게 되도 cwnd이 1로 초기화되는 slow start를 수행하므로 상당한 전송을 감소로 이어진다. 그림 6에서 보면 0.2초에서 Timeout이 발생되어짐을 볼 수가 있다. 이후 정상적인 평균 전송율에 도달 하기 까지 약 0.9초에 시간이 필요함을 알 수 있다. 이 경우 송신단의 TCP flow에 경쟁에 의해 송신단의 data 패킷이 drop되는 것을 볼 수 있는데 이를 위해서 다운 링크의 Queueing 방식을 FQ(Fair Queue)방식을 사용하였다. 큐잉 방식을 달리한 후, 제안한 TCP를 사용하여 시뮬레이션을 해보았다. 그 결과는 그림 6-3에서 나타내고 있다.

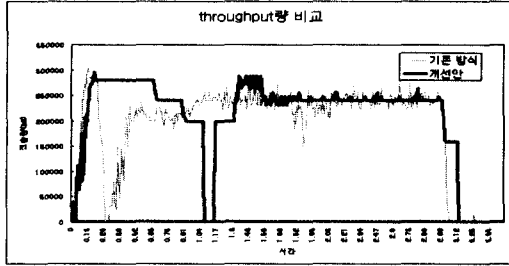


그림 8 다중 분할 방식에서 성능비교

1.02초경에 수신단의 전송 버퍼의 오버플로우가 발생되어 수신단이 TCP헤더의 FUL bit를 1로 셋팅 하여 송신단이 cwnd를 고정 한채 패킷이 전송되지 않다가 수신단의 전송버퍼의 빈공간이 확보된 시점에 FUL bit가 0으로 셋팅된 ACK 패킷을 전송하여 송신단이 이전의 cwnd로 전송함을 볼 수 있다. 시뮬레이션한 결과, 제안된 TCP가 14.57%의 전송율을 증가를 보여주고 있다.

제안된 TCP의 성능을 알아보기 위해 각각 수신단의 전송 버퍼에서 drop된 ACK수를 비교하여 보았다. 그림 9는 이를 나타내고 있다.

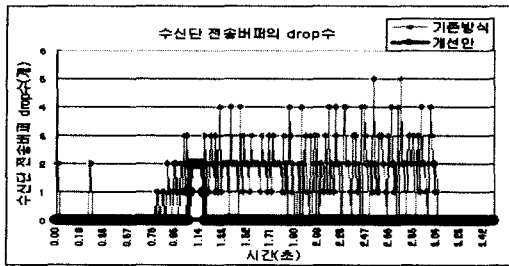


그림 9 수신단의 전송버퍼 손실된 패킷수

그림 8과 같이 제안된 TCP에서는 1.02초에 전송 버퍼의 오버플로우가 발생된다. 하지만 그 이후로는 전송 버퍼에서 ACK의 손실을 발생하지 않는다. 기존 TCP의 수신단의 전송 버퍼의 ACK 손실률은 5.24%이고 제안된 TCP는 0.05%의 ACK 손실률을 가진다.

다음으로 3초까지 FTP를 이용한 데이터 전송을 수행하였을 때의 시간에 따르는 송신단에서의 전송 패킷의 일련 번호를 살펴보았다. 기존 방식의 경우는 그림 10에, 개선안의 경우는 그림 11에 각각 나타내었다. 세로축은 TCP 송신단이 전송하는 패킷의 일련 번호이고 가로축은 초 단위의 시간 축이다.

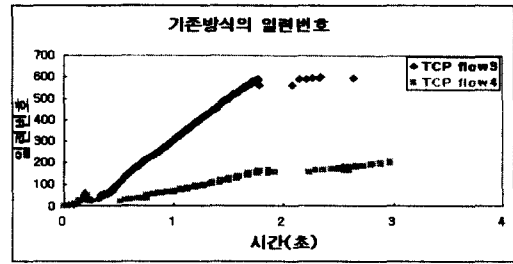


그림 10 기존 방식의 일련번호

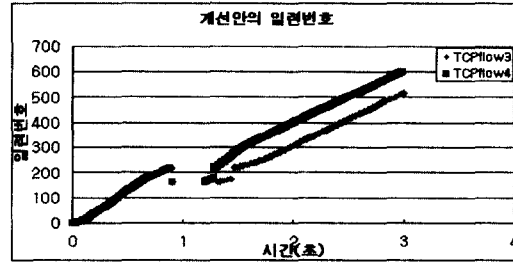


그림 11 개선안의 일련번호

기존 방식의 경우, TCP flow3의 전송율은 596packet / 3sec = 198.7packet / sec이고, TCP flow4의 전송율은 205packet / 3sec = 68.3 packet / sec 이 된다. 기존방식을 이용시 한 TCP flow들이 초기 전송시 공정성을 잃은 채 전송되어 짐을 볼 수 있다. 개선안의 경우, TCP flow3의 전송율은 600packet / 3sec = 200.0 packet / sec이고, TCP flow4의 전송율은 517packet / 3sec = 172.3 packet/ sec으로 초기 전송시부터 공정성을 가지며 높은 전송율을 보임을 알 수 있다. 개선안이 기존 방식보다 전송율이 높아지는 이유는 패킷 손실율이 낮아지고 recovery 속도가 빠르기 때문이다. 이러한 패킷 손실율이 낮아지는 것은 그림 9를 통해서도 알 수 있다.

7. 결론

본 논문에서는 비대칭 링크 상에서 TCP 전송의 문제점을 살펴 보고, 단점을 보완하기 위한 기법을 제안 하였다. 비대칭 링크에서 TCP 전송의 문제점은 수신단의 전송 버퍼의 오버플로우에 의한 것이었다. 이를 방지하기 위해 수신단에서 전송 버퍼의 오버플로우가 발생했을 때, 이를 송신단에게 통보하여 송신단은 전송을 중단하고, 혼잡 윈도우를 유지함으로써, 연결이 재개 되었을 때 송신단이 즉시 전송을 재개할 수 있도록 하였다. 실험결과 단일 TCP흐름에서는 8M/64k링크에서 13.31%, 다중 분할 방식에서 8M/384k링크에서 14.57%의 전송율이 증가하였다.

참고문헌

- [1]V.Jacobson, "Congestion Avoidance and Control", *ACM SIGCOMM'88*, August 1988.
- [2]W.R.Stevens, "TCP/IP Illustrate", Vol.1 Addison-Wesley, Nov.1994.
- [3]Douglas E.Comer and David L.Stevens, "Internetworking with TCP/IP", Prentice Hall, 1995.
- [4]V.Jacobson and R.T.Bradan, "TCP Extensions for Long Delay Paths", RFC 1072, Oct.1998.
- [5]Steven Mccune and Sally Floyd, NS (Network Simulator), <http://www.isi.edu/nsnam/ns>, 1995.