

인터넷 프로토콜의 성능향상을 위한 TCP 파라미터 조정

김명희*, 육동철*, 박봉주*, 류근열**, 박승섭
*부경대학교 전자계산학과
**위덕대학교 멀티미디어 공학과
mh02@unicorn.pknu.ac.kr

Adjusting TCP Parameter for Improving Internet Protocol Performance

Myong-Hee Kim*, Dong-Chul Yuk, Bong-Ju Park*, Kwon Ryel Ryu**, Seung-Seob Park*
Dept of Computer Science, PuKyong National University
**Dept of Multimedia Engineering, UIDUK University

요약

초고속 인터넷의 멀티미디어 서비스 통합에 대한 해결책으로 ATM 네트워크 기술이 널리 사용되고 있다. ATM 서비스를 기반으로 하는 인터넷 프로토콜에서는 ATM 층에서 하나의 셀 손실은 전체 패킷 손실을 초래하여 TCP의 성능이 저하된다. 따라서 혼잡 시 패킷의 손실을 줄이기 위해 UBR 서비스에서는 EPD 방식이, ABR 서비스에서는 ER 방식이 제안되었다.

본 논문에서는, UBR과 ABR 스위치 방식에 따른 재전송과 회복(FRR)과 Nagle 알고리즘의 사용이 성능 향상에 미치는 영향을 TCP 파라미터인 MTU 크기와 LAN 또는 WAN 환경에서 시뮬레이션 평가하였다.

1. 서론

B-ISDN는 인터넷에서 제공하는 모든 서비스를 수용할 수 있을 뿐만 아니라 광대역 및 대용량의 대역폭을 필요로 하는 서비스인 멀티미디어 서비스, 동화상 회의 등의 서비스까지 제공한다. 이를 가능하게 하기 위해 B-ISDN은 ATM을 채택하였다. 이 방식은 멀티미디어 서비스의 다양한 QoS를 제공한다. ATM Forum TM 4.0[1]에서는 ATM 트래픽을 CBR (Constant Bit Rate), rt-VBR (real time Variable Bit Rate), nrt-VBR (non real time Variable Bit Rate), ABR (Available Bit Rate), UBR (Unspecified Bit Rate)의 5가지 종류의 서비스로 정의하고 있다. 비 실시간 서비스의 예로는 데이터 전송이 있는데, ABR과 UBR 서비스가 여기에 속한다.

TCP는 현재 데이터를 전송하기 위한 인터넷 전송 프로토콜로 가장 많이 사용된다. 위의 (그림 1)은 TCP와 ATM의 프로토콜 스택을 보여주고 있는데, ATM 상에서 하나의 셀 손실은 전체 패킷 손실을 초래하여 TCP의 성능을 저하시킨다.

본 논문의 구성은 서론에 이어, 2장에서는 TCP 프로토콜의 발전과 TCP 흐름 제어에 대해 기술하였고, 3장에서는 ATM 층의 서비스 중 UBR과 ABR 서비스에 대해 간단히 서술하였고, 4장에서는 시뮬레이션 환경을 설명하였고, 마지막으로 5장에서는 본 논문의 결론을 나타내었다.

2. TCP 프로토콜의 발전

TCP는 연결지향 프로토콜로서 현재 데이터를 전송하기 위한 전송 프로토콜로 많이 사용되고 있으며

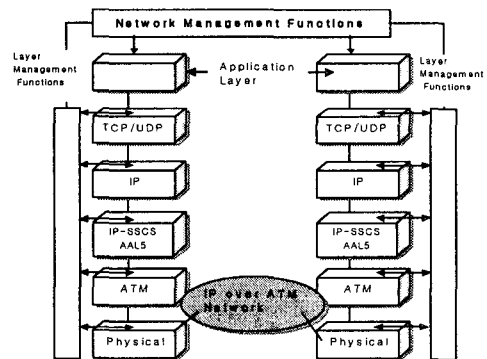


그림 1. ATM상의 TCP 프로토콜 스택

데이터를 신뢰성 있게 전송하기 위하여 윈도우를 이용하여 폭주 제어와 오류제어를 수행한다.

현재의 TCP는 1981년에 RFC 793에서 공식적으로 정의되었으면 수많은 TCP의 성능개선과 구현규격이 계속 추가되었고, 이는 1989년 RFC 1122에서 문서화되었으며 이의 확장인 RFC 1323이 1992년에 정의되었으며, <표 1>은 구현과 표준화 상태에서 TCP 흐름 제어의 발전을 보여준다.

Silly Window Syndrome Avoidance(SWS : Nagle's Algorithm)는 작은 크기의 세그먼트를 전송하는 동안 TCP 윈도우를 작게 증가하는 특징을 지닌다. 작은 데이터 세그먼트를 보낼 때 TCP와 IP 헤더가 네트워크 대역폭을 낭비하기 때문에 TCP 처리율이 낮아진다. 이를 피하기 위해 수신측에서는 작은 크기의 세그먼트가 수신될 때, 윈도우가 작게 증가하는 것을 피하고 송신측은 수신측의 데이터 세그먼트를 받아들일

표 1. TCP 흐름 제어의 발전

Objectives	Implementation	Standardization
-Connectivity -Reliability	4.2 BSD (1983) -Window flow control -Time-out based retransmission	RFC 793 (1981)
-Connectivity mitigation	4.3 BSD (1989) -Slow start -Congestion avoidance -Fast retransmit	RFC 1122 (1982)
-Fast error recovery	4.3 BSD Reno(1990) -Fast recovery	RFC 2001(1997)
-Long fat pipe	4.4 BSD (1993) -Window scale option -Time stamp option	RFC1323 (1992)
-Effective error recovery	Selective Ack	RFC 2018 (1996)

여유공간이 있더라도 수신측으로 작은 크기의 데이터 세그먼트를 보내는 것을 피해야 한다. 만일 TCP가 올려진 데이터마다 전송된다면, 1 Octet 데이터 세그먼트의 스트림이 될 것이다. 이는 TCP와 IP 헤더가 각각 20 Octet인 것을 가만할 때, 큰 대역폭의 손실을 가져 올 것이다. Nagle's Algorithm은 송신측에서 부정응답 데이터가 있거나 TCP가 MSS 세그먼트를 보낼 수 있을 때까지 또는 보내진 데이터의 ACK를 받을 때까지 모든 데이터를 기억한다[8].

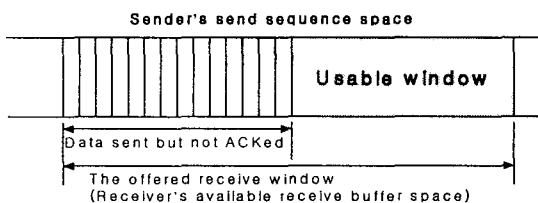


그림 2. 송신측의 사용 가능한 윈도우

3. ATM 층의 서비스

ATM은 연결형 방식에 기반을 두고 있으면 중단 시스템 간에 데이터를 전송하기 위해서 만들어진 연결을 가상연결이라 부른다. VC를 통과하는 데이터를 53 옥텟 크기의 셀로 분할되어 전송된다.

ATM 망에서 제공하는 UBR 서비스는 명백한 혼잡 제어 메커니즘이 없다. 이러한 단점을 보완한 셀 폐기 정책 중 하나인 EPD (Early Packet Discard)를 이용해 망에서 불필요한 대역폭 낭비를 줄일 수 있다[4][6]. ABR 서비스는 고속의 데이터 전송을 위한 서비스로 적은 셀 손실과 시간에 따라 변하는 망의 자원을 효율적으로 사용할 수 있는 능력 때문에 데이터 트래픽을 전송하기에 적합한 서비스이다. 이것은 RM 셀을 이용하여 폭주제어를 하고 제어 변수들을 적절하게 설정함으로써 셀 손실 없는 데이터 전

송을 가능하도록 송신측의 전송속도를 조절한다[5].

4. 시뮬레이션 환경

본 장은 시뮬레이션 망 모델과 파라미터에 대해 설명한다. C++로 작성된 YATS ATM 시뮬레이션 도구 [3]를 사용하여 시뮬레이션 하였다.

4.1 시뮬레이션 모델

UBR과 ABR 서비스 상에서 TCP의 성능 분석을 위해 (그림 3)와 같은 하나의 병목 구간을 가지는 peer-to-peer 모델을 설정하였다. 전송방향은 단방향 트래픽으로 데이터를 전송하게 하는 모델이다[9][10]. <표 1>에서와 같이 컨택션 수가 늘어남에 따라 처리율과 공정성이 저하됨을 보여주고 있다. 그러나 EPD 스위치 알고리즘의 경우 컨택션 수가 50 일 때 높은 처리율을 나타내어 이후 실험부터는 컨택션 수를 50으로 제한한다.

표 2. VC수에 따른 처리율과 공정성

방식 VC수	EPD		EFCI		ER	
	처리율	공정성	처리율	공정성	처리율	공정성
10	82.4	82.5	126.2	100.0	126.2	100.0
20	103.0	81.0	126.1	97.6	126.2	100.0
30	109.0	76.6	106.3	83.6	106.3	100.0
40	106.2	66.6	107.7	85.2	107.7	100.0
50	128.2	62.6	105.4	79.5	105.4	100.0
60	116.9	66.0	95.6	61.4	95.5	100.0
70	126.5	59.6	97.0	51.4	97.3	100.0

* 처리율; Mbps, 공정성; 백분율

스위치 방식으로는 ABR 스위치 알고리즘인 EFCI, ER과 UBR 스위치 알고리즘인 EPD 알고리즘을 사용하여 TCP 성능분석을 하였고, 특히 TCP 파라미터인 Nagle's Algorithm과 FRR은 ABR과 UBR 상에서의 TCP 성능 향상에 미치는 영향을 관찰하였다. 그리고 송신측의 TCP 애플리케이션은 고정 프레임 길이를 가지면서 지속적으로 프레임 전송하였다.

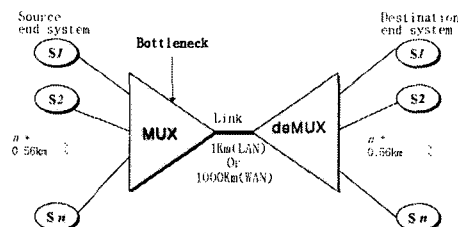


그림 3. 시뮬레이션 모델

4.2 시뮬레이션 파라미터 설정 및 분석

각 ABR, UBR, TCP 컨넥션, 스위치, Link의 기본 파라미터 설정은 <표 3>과 같이 나타내었다. 본 논문에서 TCP의 파라미터를 조정하여 ATM 상에서의 TCP 성능개선을 도모하였다. 다시 말해서, 정의된 파라미터 값으로써 성능 개선을 위해 FRR과 Nagle's Algorithm을 ON/OFF 하여 각각의 알고리즘이 MTU 크기나 LAN 또는 WAN 환경에서 TCP 성능에 미치는 영향을 관찰하였다.

표 3. 시뮬레이션 파라미터

Component	Parameter	Value
TCP application	Data Frame	4 Kbytes
TCP	Mean Packet Processing	300 us
	Send/Receiver buffer size	64 Kbytes
	Maximum segment size	2 Kbytes
	Maximum receiver window size	64 Kbytes
	Fast retransmission and recovery Nagle	on/off on/off
Link	speed between switch and end system	150 Mbps n * 0.56 km (n: no. of VC)
	Distance : LAN or WAN	1 or 1000km
ABR end system	PCR	150 Mbps
	Nrm	32
	MCR	0 Mbps
ABR/UBR switch	Output Buffer Size	4096 cells
	High threshold	300 cells
	Low threshold	200 cells
	EPD Threshold	3500 cells

성능 평가 요소로서는 처리율과 공정성을 기준으로 하였다. 여기서 TCP 처리율은 목적지 TCP 층에서 측정되고, 목적지 애플리케이션으로 전달된 총 바이트 수로 정의되며, 이 값은 전체 시뮬레이션 시간(300 ms)으로 나눈 값이다.

ATM 계층에서는 TCP 최대 세그먼트 크기(MSS)를 2000 바이트에 TCP 헤더 20 바이트, IP 헤더 20 바이트, LLC 헤더 8 바이트, AAL 트래일러 8 바이트에 패딩과 ATM 층의 헤더 5 바이트가 합쳐져서 총 43 셀이 된다. 그래서 실제로 이용 가능한 최대 처리율은 $2000 \div 2279 = 0.87$ 가 된다. 즉, 56 바이트의 오버헤드로 인해 최대 전송 가능한 TCP 처리율은 150 Mbps가 아니라 130.5 Mbps가 된다.

공정성은 서로 다른 VC 간 대역폭 할당에 있어서 공평하게 할당받았는가를 평가하는 기준으로 다음 식과 같이 주어진다[5].

$$Fairness = \frac{(\sum x_i)^2}{n \times (\sum x_i^2)}$$

여기서, $x_i = T_i / O_i$ 로 Y_i ($i = 1 \dots n$)는 각 VC 마다 성공적으로 도착된 패킷의 측정된 처리값으로 단위시간 당 전송 받은 양을 바이트로 계산하였고, O_i ($i = 1 \dots n$)

는 최대-최소의 최적 처리율이고, n 은 전체 VC의 수이다. 여기에서 공정성의 기준값은 1로 설정하였는데, 공정성의 의미는 각 TCP 컨넥션 간의 대역폭을 공평하게 할당받았는가를 평가한다.

(그림 4)는 처리율을 나타내는데 ER 알고리즘은 FRR과 Nagle을 ON 했을 때와 하지 않았을 때의 처리율이 같았고, EFCI 알고리즘은 Nagle에는 전혀 반응을 보이

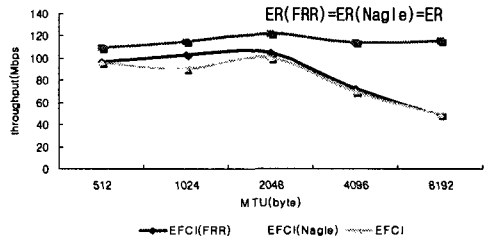


그림 4. ABR 스위치의 처리율

지 않은 반면에 FRR에는 성능이 향상됨을 보였다. 특히 MTU 크기가 2048 바이트를 넘어설 때 성능이 급격히 저하되는 것을 볼 수 있다. 이와 같은 결과는, ER 스위치는 ABR을 흐름제어에 의해 전송량을 할당받아 ACR을 조절하기 때문에 이들 TCP 파라미터에 영향을 받지 않는다. 그렇지만 EFCI 스위치는 폭주 시 RM 셀의 EFCI 비트에 마크하는 것으로 전송량을 조절하므로 성능을 변화를 가져올 수 있었다.

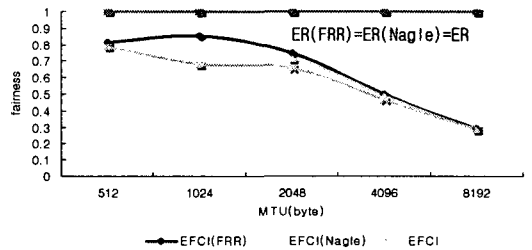


그림 5. ABR 스위치의 공정성

(그림 5)에서는 EFCI가 처리율에서는 Nagle에 반응 보이지 않은 반면 공정성 면에서는 약간 향상되었다.

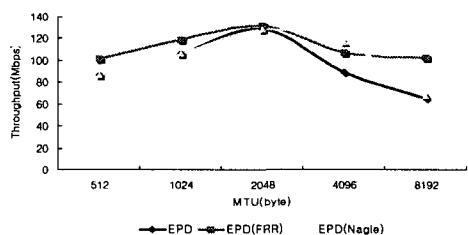


그림 6. UBR 스위치의 공정성

(그림 6)은 EPD 스위치의 처리율을 나타내고 있는데, ABR 스위치와 달리 FRR이나 Nagle을 ON 했을 때 성능이 향상됨을 보여주고 있다. 이는 UBR 서비스가 명백한 제어 메커니즘을 지니지 않기 때문에 상위계층의 제어를 받기 때문이다.

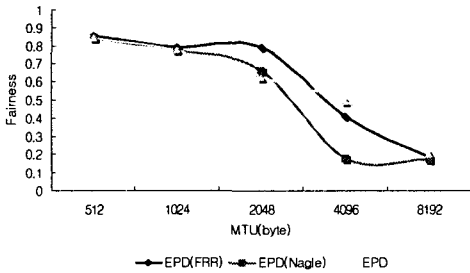


그림 7. UBR 스위치의 공정성

(그림 7)은 공정성을 보여주고 있는데, 공정성 면에서는 TCP 파라미터를 사용하지 않았을 때 더 나은 것을 확인 할 수 있다. 처리율이나 공정성 모두 ABR 상에서의 성능과 마찬가지로 MTU 크기가 2048 바이트에서 높았다.

<표 4>는 이들 TCP 파라미터들이 거리에 따라서는 어떻게 변화하는지를 보여주고 있다. 거리는 크게 LAN 일 때와 WAN 일 때로 나누어서 실험하였는데, LAN은 스위치간 거리가 1 Km로 하고, WAN 일 때는 1000 Km로 가정하였다.

표 4. FRR과 Nagle 알고리즘의 효과

	EPD		EFIC		ER	
	처리율	공정성	처리율	공정성	처리율	공정성
LAN	120.6	66.9	104.5	75.0	121.4	100.0
WAN	114.6	70.8	75.2	75.4	120.2	100.0
Nagle	126.8	68.6	104.0	79.3	121.4	100.0
WAN	123.7	60.7	75.2	75.4	120.2	100.0
FRR	130.3	82.6	104.8	75.3	121.4	100.0
WAN	131.7	80.8	98.4	82.9	120.2	100.0

LAN과 WAN 환경에서 FRR과 Nagle을 ON/OFF 했을 때 처리율과 공정성을 나타내었다. 그 결과, EPD는 앞의 실험 결과와 같이 TCP 파라미터를 사용했을 때 성능의 향상을 보였고 ER은 성능의 변화를 관찰 할 수 없었다.

또한 EPD의 경우, LAN 환경에서 FRR과 Nagle 모두 성능향상을 가져오지만 WAN 환경에서 LAN 환경과 비교해서 낮은 성능을 보였다. 특히 FRR은 TCP층에서 혼잡 발생 시 타임아웃 때까지 기다리지 않고 잘못된 패킷을 재전송 하는 기법이 거리가 스위치간 거리가 가까운 LAN 환경에서 UBR의 셀 손실을 막는데 도움이 되지만 상대적으로 거리가 먼 WAN 환경에서는 성능 향

상을 기대하기 어렵다. 또 Nagle 알고리즘 보다는 FRR이 성능 향상에 미치는 영향을 큼을 알 수 있었다.

5. 결론

본 논문에서는 시뮬레이션의 결과를 통해 다음과 같은 결론을 얻었다.

- 1) ATM 상에서의 TCP 성능이 노드 수에 따라서 좌우됨을 확인 할 수 있었다.
- 2) UBR의 EPD 스위치는 TCP층의 파라미터인 빠른 재전송과 회복(FRR)와 Nagle's Algorithm이 UBR 상의 TCP 성능을 개선하는데 중요한 요소인 것을 알 수 있었다.
- 3) TCP 중단 시스템 파라미터들은 LAN 환경에서 보다 WAN 환경에서는 처리율과 공정성 면에서 성능저하를 가져온다 것을 확인 할 수 있었다.

참고문헌

- [1] ATM Forum, "ATM Traffic Management Specification Version 4.0", Apr. 1996.
- [2] M. Mathis et al., "TCP Acknowledgment Options", RFC 2018, October 1996.
- [3] YATS simulation for ATM Networks, Dresden University Technology, 1997.
- [4] R. Goyal, et al., "UBR+:Improving Performance and modelling and analysis volume 3", 1st ED., Chapman & Hall, 1997.
- [5] Sufian YOUSEF and Caroline STRANGE "TCP/IP over Challenges in Enterprise Network integration", IEEE International Conference on ATM , ICATM'98, 1998, pp. 447-453.
- [6] Rohit Goyal, Raj Jain, Shiv Kalyanaraman, Sonia Fahmy, Bobby Vandalore, "Improving the performance of TCP over the ATM-UBR service", Computer Communications, V.21 N.10, 1998.
- [7] Masatoshi Kawarasaki, Mika Ishizuka and Arata Koike, "Dynamics of TCP flow control over High-Speed ATM Networks", IEEE, 1998.
- [8] Douglas E. Comer and John C. Lin, "TCP Buffering AND performance Over An ATM Network", Journal of Internetworking : Research and Experience, 6(1):1-13, March. 1995.
- [9] 김명희, 오정순, 박봉주, 박승섭, "ATM 망에서 ABR/UBR 서비스상의 TCP 성능에 대한 평가", 제 13회 한국 정보처리 학회 춘계 학술대회, no.37, 2000년 4월.
- [10] Myong-Hee Kim, Dong-Cheol Yuk, Dong-Ryel Ryu, Seung-Seob Park, "The performance evaluation of Internet protocol with data traffic over ATM network" EALIT2000, 8. 2000, pp.262-268.