

ATM 네트워크상에서 TCP트래픽에 관한 시뮬레이션 평가

0

박봉주, 육동철, 박승섭
부경대학교 대학원 전자계산학과

Simulation Evaluation for TCP Traffic over ATM Network

0

Bong-Ju Park, Dong-Cheol Yuk, Seung-Seob Park
Dept. of Computer Science, Graduate school, Pukyong Nat'l University

요 약

초고속통신의 실현과 다양한 멀티미디어 서비스를 위한 TCP/IP통신의 전송 메카니즘으로서 ATM이 널리 사용되고 있다. TCP층과 ATM층의 상호작용이 TCP층뿐만 아니라 ATM 층에도 영향을 미쳐 처리율과 공정성의 성능이 저하되므로, ATM상의 TCP 트래픽 파라미터, 즉 TG(timer granularity)와 윈도우 크기를 적절하게 조정함으로써 TCP 흐름제어가 ABR 율제어와 상호작용하지 않는다. 그 결과로 상호작용의 영향을 최소화 할 수있으며, TCP의 성능 향상을 가져올 수 있다.

본 논문에서는 윈도우 크기와 TG와 같은 TCP 파라미터를 조절하여 VBR서비스 상에서의 TCP 트래픽 성능평가를 통해 높은 처리율과 공정성을 나타내었다.

1. 서론

ATM(Asynchronous Transfer Mod)은 B-ISDN을 실현하기 위한 기본 방식으로 채택되어 탄생한 것이다. ATM Forum TM(Traffic Management) 4.0은 높은 처리율, 효율적인 자원활용, 사용자의 다양한 요구 만족을 위해 트래픽 특성과 서비스 품질에 따라 5개의 서비스 부류를 정의한다[1]. 실시간 서비스는 CBR(Constant Bit Rate), rt-VBR(real-time Variable Bit Rate)로 구성되며, 비실시간 서비스는 nrt-VBR(ont real-tim Variable Bit Rate), ABR(Availiable Bit Rate), UBR(Unspecified Bit Rate)로 구성된다.

ABR 서비스는 CBR/VBR의 실시간 서비스를 위한 대역폭을 제외한 나머지 대역폭을 ABR 접속에 할당함으로써 전송 링크의 전체 대역폭 활용도를 높이는 것을 목적으로 한다. ABR 서비스는 긴 셀 지연이 문제되지 않으며 망의 상태에 따라 신속하게 적응할 수 있는 비 실시간 데이터 전송에 가장 적합하다. 혼잡제어는 ABR 서비스에서 매우 중요한 부분을 차지한다. 또한 혼잡제어는 전송 링크를 공유하는 ABR

접속간에 공정하게 대역폭을 할당한다.

UBR 서비스는 상위 어떤 유용한 대역폭을 사용하고 셀 지연에 민감하지 않는 트래픽을 전송하는데 유리하며, 특히 ATM 망에서 제공하는 UBR 서비스는 명백한 혼잡 제어 메커니즘이 없다. 이러한 단점을 보완한 셀 폐기 정책 중 하나인 EPD(Early Packet Discard)를 이용해서 망의 불필요한 대역폭 낭비를 줄일 수 있다.

TCP는 현재 데이터를 전송하기 위한 전송 프로토콜로 가장 많이 사용된다. 기본적으로 데이터를 신뢰성 있게 전송하기 위하여 윈도우를 이용하며 Slow Start, 폭주회피, 그리고 재전송 단계로 구성된 폭주 제어와 오류제어를 수행한다[2]. TCP의 윈도우 기반 제어와 ABR 전송을 기반 제어가 독립적으로 동작한다.

TCP는 하위계층의 망 정보를 이용하지 않고 응답 메시지를 이용하여 망의 상태를 예측하기 때문에 갑작스런 트래픽 양의 증가는 망에서의 지연 변이를 급격히 증가시키고, TCP에서 잘못된 재전송을 유발할 가능성이 커진다. 이로 인해 대역폭의 낭비와 처

리율의 저하를 가져온다. 이러한 문제는 ATM 하부구조하에서 효과적인 TCP 동작의 연구가 진행 중에 있다[3].

본 논문에서는 효과적인 처리율과 공정성의 관점에서 ATM 서비스 상에서의 TCP 트래픽의 성능향상을 위해 주요 파라미터인 TG와 윈도우 크기를 제어하며 ERICA, UBR, UBR+EPD, EFCI 각 알고리즘에 대해 시뮬레이션을 통한 성능 평가를 하였다. 본 논문의 구성은 1장 서론에 이어, 2장에서는 TCP over ABR/UBR 서비스를 기술하고, 3장에서 ATM 계층에서의 혼잡 제어 방식, 4장에서는 시뮬레이션 환경 및 결과를 서술하였고, 마지막으로 5장에서는 논문의 결론을 서술하였다.

2. TCP트래픽에서의 ABR 서비스

2.1 TCP트래픽에서 ABR

TCP가 ABR서비스 상에서 동작하게 될 경우 TCP의 윈도우 기반 흐름제어는 ABR 서비스의 전송률 기반 폭주 제어 위에서 동작하게 되는데 이는 성능의 저하를 가져올 수 있다[4]. 이러한 이유는 (그림 1)과 같이 TCP의 윈도우 기반 제어와 ABR 서비스의 전송률 기반 제어가 독립적으로 동작하기 때문이다. TCP는 하위계층의 망 정보를 이용하지 않고 응답 메시

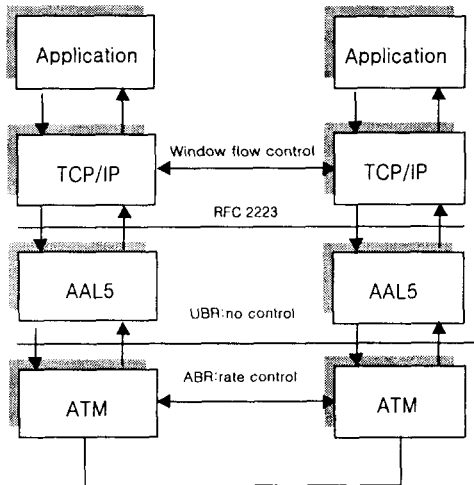


그림 1. 서비스 처리 범주와 프로토콜 계층

지를 이용하여 망의 상태를 예측하기 때문에 갑작스런 트래픽 양의 증가는 망에서의 지연 변이를 급격

히 증가시켜 TCP에서 잘못된 재전송을 일으킬 가능성이 있다. 이런 잘못된 재전송은 정상적으로 전송된 셀을 다시 전송 하게 되므로 대역폭의 낭비와 처리율의 저하를 초래한다. TCP와 ABR의 상호 작용을 방지하기 위해서 coarse TG와 큰 윈도우 사이즈가 요구된다. TCP에서는 타이머가 만료될 때까지 ACK가 수신되지 않으면 전송한 세그먼트를 다시 전송하는 재전송방법을 사용하여 TCP 송신측은 최대 윈도우 크기만큼의 세그먼트를 ACK없이 송신할 수 있다. 송신된 세그먼트에는 데이터 세그먼트의 순서 번호를 가지고 있다고 가정한다. ACK를 수신한 송신측은 할당받은 크레딧에 따라서 ACK 없이 송신할 수 있는 자신의 윈도우를 증가시킨다[6].

2.2 TCP 혼잡제어

TCP의 혼잡제어로는 Slow Start, 폭주회피, 그리고 재전송 단계 알고리즘을 사용한다[7]. (그림2)와 같이 SS(Slow Start)는 송신원에서 수신원에 연결 요청 수락 후, 초기 데이터를 전송하고 수신측 TCP 컨넥션은 송신원에 ACK를 보내서 수신에 성공했음을 알린다. 이때 송신원은 RTT(Round Trip Time)를 계산하고, RTT를 기본으로 RTO를 계산한다.

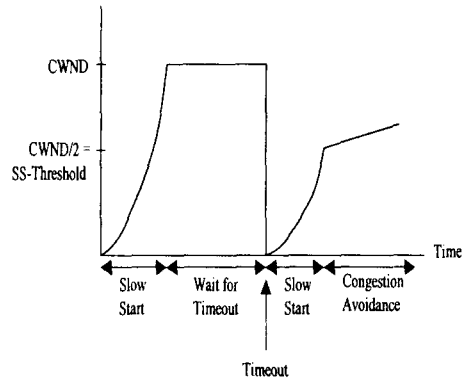


그림 2. Slow start와 폭주회피 단계의 폭주윈도우의 크기 변화

SS동안에는 매 ACK에 대해서 폭주 윈도우 (congestion window: cwnd)의 크기를 하나씩 증가시키며, 폭주 윈도우의 크기 증가는 SST(Slow Start Threshold)에 이르거나 세그먼트의 손실이 감지될 때까지 계속된다. 송신원은 중복된 ACK가 n 개 이상

수신되면 손실로 간주하여 재전송 한다. 이것은 재전송 타이머가 만료될 때까지 기다려야 하는 시간을 줄이는 방법이다[5].

3. ATM 계층에서의 혼잡 제어 방식

3.1 UBR과 UBR+EPD상에서의 동작

UBR 서비스는 CBR, VBR, ABR 서비스가 사용하고 남은 대역폭을 사용하는 지연에 민감하지 않는 트래픽을 전송하는데 유리한 서비스이다. 상위 계층인 TCP 제어방식에 의존한다. ATM 스위치의 버퍼에서 하나의 셀 손실은 TCP 층의 하나의 패킷 손실로 이어진다. 이는 망에서 낮은 처리율과 공정성으로 이어져 확실한 QoS의 보장이 어렵다. 이러한 단점을 보완한 셀 폐기 정책인 EPD(Early Packet Discard)나 PPD(Partial Packet Discard), RED(Random Early Detection)[8] 등을 사용해 망에 불필요한 대역폭 낭비를 줄일 수 있다. EPD는 버퍼에 입계치를 두어 이 입계치를 넘는 패킷의 첫 번째 셀이 있으면 그 첫 번째 셀과 같은 가상채널 식별자를 가지는 패킷 전부를 폐기하는 방법이다. 버퍼 오버플로우로 야기되는 패킷의 손실을 방지하여 불필요한 재전송과 대역폭 낭비를 막는다.

3.2 ABR 상에서의 동작

ABR 트래픽에서는 망의 상태에 따라 송신측의 전송률을 조정할 수 있고, 이용 가능한 네트워크 자원을 공평하고 효율적으로 사용할 수 있다. 또한, 매우 버스티한 데이터 어플리케이션을 지원하기 위해 제안되었다. RM 셀은 제어정보에 대한 자세한 사항을 기술하여 소스로 전달하고 소스는 이 정보 셀을 이용하여 현재의 네트워크에서 적합한 셀 전송율을 조정하게 된다.

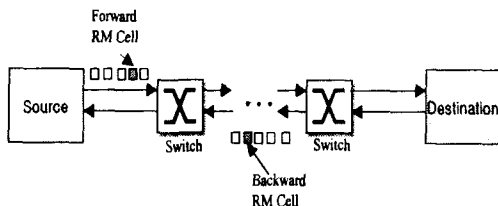


그림 4. RM 셀의 경로

(그림4)와 같이 전송률 기반이라 송신측이 정해진 전송률로 데이터를 보낸다는 의미이다. 지금까지 제안된 대표적인 전송률 기반의 ABR 스위치로는 크게 EFCI(Explicit Forward Congestion Indication)스위치와 ER(Explicit Rate) 스위치로 나눌 수 있다. ER 스위치는 VC간의 링크 전송률의 공평한 몫인 공정성을 계산하여 역방향 RM 셀의 ER 필드에 실어 송신원에 알리게 된다. 이렇게 함으로써 송신측은 경로상의 모든 스위치의 허가된 MCR(Minimum Cell Rate) 값을 받게 되고 병목현상을 고려할 수 있게 된다. 이러한 ER 스위치의 구조는 EPRCA(Enhanced Pro-protional Rate Congestion Avoidance), ERICA(Explicit Rate Indication for congestion Avoidance), ERICA+, CAPC등의 구조가 있다[5].

4. 시뮬레이션 환경

본 장은 시뮬레이션 망 모델과 파라미터에 대해 설명하고 결과를 분석한다. C++로 작성된 YATS ATM 시뮬레이션 도구[9]를 사용하여 시뮬레이션 하였다.

4.1 시뮬레이션 모델

VBR 자원이 있는 경우 ATM상에서 TCP의 성능 분석을 위해 (그림 5)과 같은 하나의 병목 구간을 가지는 피어 투 피어 모델을 설정하였다. 전송방향은 단방향 트래픽으로만 데이터를 전송하게 한 모델이다. 링크 속도는 150Mbps이고, 스위치간 거리는 56Km, TCP 컨넥션 수는 55개로 제한한다.

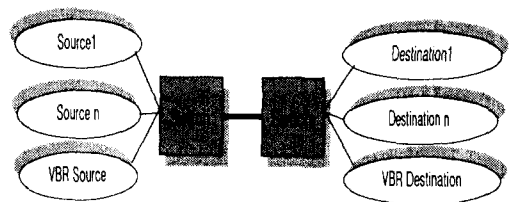


그림 5. 망 구성도

스위치 방식으로는 EFCI, ERICA를 적용해서 ABR 상에서의 TCP 성능을 분석하였고, UBR 상에서의 TCP 성능 분석을 위해서는 UBR 그 자체와 UBR+EPD 알고리즘을 적용하였다.

4.2 시뮬레이션 파라미터 설정 및 결과

각 ABR, UBR, VBR, TCP 컨택션, 스위치, Link의 기본 파라미터 설정은 아래 <표 1>과 같이 나타내었다. 성능 평가 요소로서는 TG와 윈도우크기로 하였다. 공정성은 서로 다른 VC 간 대역폭 할당에 있어서 공평하게 할당받았는가를 평가하는 기준으로 다음 식과 같이 주어진다.

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

표 1. 시뮬레이션 파라미터

TCP Application	
Data Frame = 4 Kbytes	
VBR	
MMBP source(ON/OFF)	
EB(ON 상태의 평균 duration) = 3000	
ED(OFF 상태의 평균 duration)= 3000	
ED(분산된 셀의 평균 간격)= 1.5 μ s	
TCP	
Mean Packet Processing = 300 μ s	
Send/Receiver Buffer Size = 64 Kbytes	
Fast Retransmission and Recovery = off	
Link	
speed = 150Mbps	
between Switch Distance = 56 Km	
ABR end system	
Packet Processing Delay = 1 μ s	
Buffer Size = 100 Kbytes	
PCR(Peak Cell rate) = 150 Mbps	
Nrm = 32, MCR = 0	
FRTT = 10 msec	
ABR Switch	
Output Buffer Size = 4096 cells	
High threshold = 300 cells	
Low Threshold = 200 cells	
UBR Switch	
Output Buffer Size = 4096 cells	
EPD Threshold = 3500 cells	

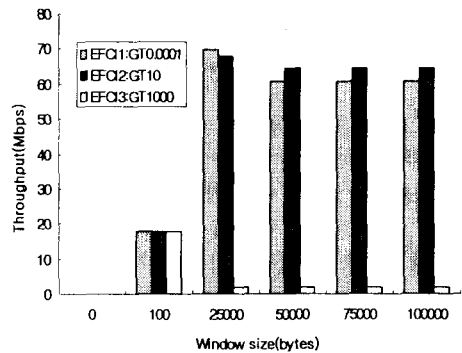
여기에서 공정성의 기준값으로 1을 설정하였다. 실험 결과는 <표 2>에서는 MTU 크기가 2048 bytes 일 때 가장 좋은 처리율을 보였지만, 공정성측면에서는 오히려 512 bytes에서 좋은 성능을 보여 주었다. (그림6)에서 TG(0.0001 msec, 10 msec, 1000 msec)와 윈도우 크기 변화에 따른 처리율과 공정성을 보여 주고 있다.

UBR+EPD에서 TCP가 가장 높은 처리율을 보이고 있다. TCP의 TG가 지나치게 커도 ATM층의 성능을 악화

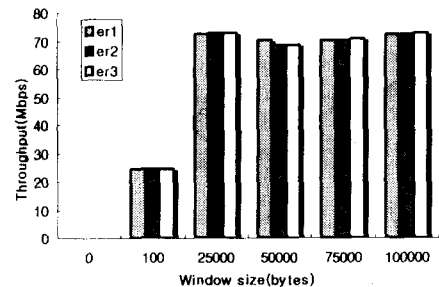
시킬 수 있음을 보여주고 있다. 실험 결과 ABR 기반의 TCP에서 더 좋은 성능과 공정성을 보임을 알 수 있다. <표 2>에서 볼 수 있듯이 MTU크기가 UBR과 ABR에 미치는 영향을 실험한 결과 처리율과 공정성에 있어서 UBR이 영향을 많이 받고 있음을 보였다. 특히 MTU크기가 2048bytes를 넘어섰을 때, 급격한 성능저하를 보였다. 공정성 측면에서도 ER을 제외한 나머지가 모두 많은 영향을 받았으며, 역시 MTU 크기가 2048bytes 이상에서 급격한 성능저하가 나타났다.

표 2. MTU 크기에 따른 처리율과 공정성

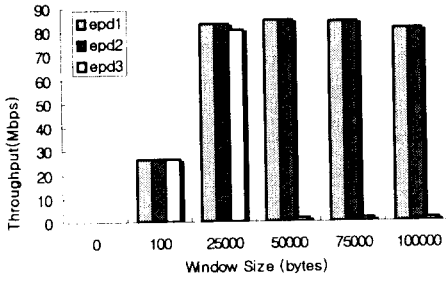
방식 MTU	UBR		EPD		EFCI		ER	
	처리율	공정성	처리율	공정성	처리율	공정성	처리율	공정성
512	53.72	0.931	68.43	0.93	57.10	0.926	64.05	0.997
1024	67.72	0.863	78.97	0.937	61.20	0.800	66.41	0.994
2048	72.57	0.756	75.45	0.901	64.42	0.898	70.98	0.980
4096	0	0	6.53	0.204	21.06	0.342	69.32	0.990
8192	0	0	67.81	0.174	40.17	0.383	65.91	0.989



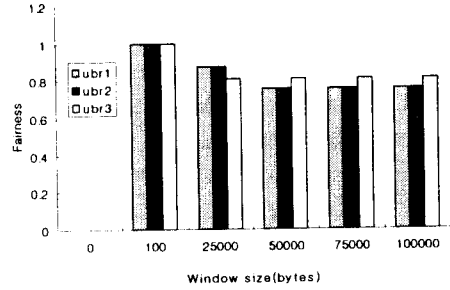
(a) EFCI 알고리즘



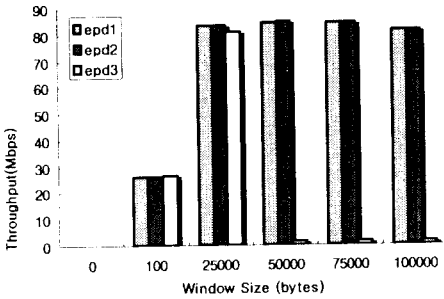
(b) ER 알고리즘



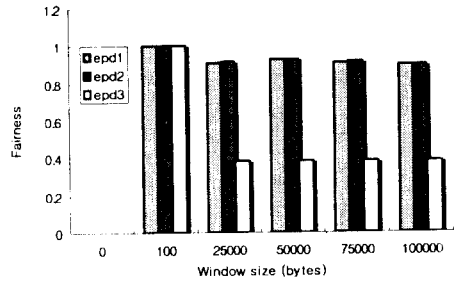
(c) UBR 알고리즘



(c) UBR 알고리즘



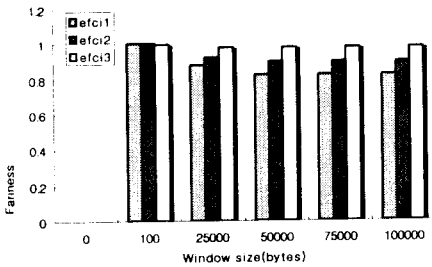
(d) EPD 알고리즘



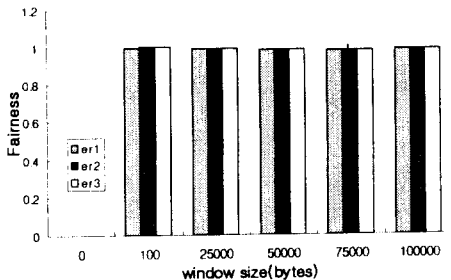
(d) EPD 알고리즘

(그림 6) TG와 윈도우크기에 따른 처리율

(그림 7) TG와 윈도우크기에 따른 공정성



(a) EFCI 알고리즘



(b) ER 알고리즘

5. 결론

본 논문에서는 VBR을 포함한 망구성 형태에서 TCP 트래픽의 성능 개선방안을 위해 기존의 여러 가지 ER 스위치 알고리즘과 UBR, UBR+EPD 알고리즘을 이용한 처리율과 공정성을 구하여 다음과 같은 결론을 얻었다.

UBR서비스상의 TCP 트래픽은 세그먼트가 커질 때, 성능이 급격히 악화된다. EPD는 TG가 지나치게 크면 처리율과 공정성이 모두 악화됨을 확인하였다.

ABR은 패킷 손실을 제거할 수 있기 때문에 ABR 기반의 TCP가 가장 좋은 성능을 보임을 알 수 있다. TCP 흐름제어가 ABR 전송률 제어와 상호 작용하지 않도록 하기 위해 TG와 MTU 크기, 윈도우 크기와 같은 TCP 파라미터가 중요한 요소임을 확인하였다.

VBR서비스상의 TCP 트래픽의 처리율이 순수 ABR/UBR서비스상의 처리율의 70%수준을 보여주었고, 공정성에 있어서는 매우 유사한 결과를 가져왔다.

참고문헌

- [1] ATM Forum "ATM Traffic Management

- Specification Version 4.0", Apr. 1996.
- [2] Janey C. Hoe : Improving the startup behavior of congestion control scheme for TCP, proceedings of the ACM SIGCOMM, 1996
- [3] Sufian YOUSEF and Caroline STRANGE "TCP/IP over Challenges in Enterprise Network intergration ", IEEE International Conference on ATM, ICATM'98 1998, pp 447-453.
- [4] Per Janhansson, Elin Wedlund, Johan M. Karlsson, "Interaction Between TCP Flow Control and ABR Rate Control", Proceedings of the IEEE ATM'97 workshop, May 1997.
- [5] Masatoshi Kawarasaki, Mika Ishizuka and Arata Koike "Dynamics of TCP flow control over High-Speed ATM Networks" , 0-7803-4788-9/98 , IEEE 1998.
- [6] V. Jacobson, "Congestion Avoidance and Control", Proceedings of the SIGCOMM'98 Symposium , pp.314-332, August. 1998.
- [7] S. Kalyanaraman, R. Jain, R. Goyal, S. Fahmy, and Vandalore, "The ERICA Switch Algorithm for ABR Traffic Management in ATM Networks", Submitted to IEEE/ACM Transactions on Networking, November 1997
- [8] O. Elloumi, H. Afifi, "RED Algorithm in ATM Networks", IEEE ATM '97 Workshop, pp.312-319, July 1997.
- [9] YATS simulation for ATM Networks, Dresden University Technology, 1997.
- [10] 박승섭, 육동철 "ABR 과 UBR 서비스상에서의 인터넷 프로토콜: 문제점, 해결방안, 그리고 성능평가", 정보처리 논문지, 제6권 제 11S 호 pp. 3260-3268, 1999년 11월.