

# RM 셀 정보를 이용한 TCP over ATM의 성능 개선

정회원 최진혁\*, 정재일\*\*

## Performance improvement of TCP over ATM using RM cell information

Jin-Hyuk Choi\*, Jae-Il Jung\*\* *Regular Members*

### 요약

ABR (Available Bit Rate) 서비스는 ATM (Asynchronous Transfer Mode) 에서 데이터 응용을 지원하기 위하여 개발되었으며, TCP (Transmission Control Protocol) 는 현재 사용되는 데이터망에서 가장 널리 이용되는 트랜스포트 계층 프로토콜이다. 따라서 TCP over ABR의 성능 향상은 대단히 중요한 문제이다. 본 논문에서는 RM (Resource Management) 셀의 정보를 이용한 TCP congestion 윈도우 크기 조절을 제안한다. 제안된 방식의 사용으로 TCP 윈도우 크기 결정에 보다 신뢰성 있는 망 정보 (RM 셀 내의 ER 값)를 사용하게 된다. 시뮬레이션은 다양한 망 상황과 다양한 망 구성에서 행하여졌으며 시뮬레이션 결과는 제안된 방식의 사용으로 TCP의 성능이 월등히 향상됨을 보인다.

### ABSTRACT

The ABR(Available Bit Rate) service has been developed to support data applications over ATM(Asynchronous Transfer Mode) networks, and TCP is the most widely used transport layer protocol in existing data networks. Thus, the performance improvement of TCP over ABR is an issue of cardinal importance. In this paper, we propose an algorithm of the adjustment of congestion window size utilizing RM (Resource Management) cell information. The motivation is to use more reliable information in ER (Explicit Rate) field within RM cell to determine TCP window size. Simulations are performed under various network situations and topologies, and the results confirm that the overall TCP performance using the suggested algorithm is improved, as compared to TCP-Reno.

### I. 서론

ATM 방식에서는 다양한 종류의 트래픽들을 지원하기 위해 4 가지 서비스 클래스들을 제공하고 있는데, 그 중 가장 복잡한 방식인 ABR (Available Bit Rate) 서비스 클래스<sup>[1]</sup>는 ATM에서 데이터 응용을 지원하기 위하여 개발되었다. ABR에서의 흐름제어는 RM (Resource Management) 셀을 이용하

여 망으로부터 얻은 feedback정보를 송신 측에 전달하여 송신 측에서 이 정보를 이용하여 전송률을 적절히 조절하는 방법으로, 크게 EFCI (Explicit forward Congestion Indication), ER (Explicit Rate) 의 두 가지 방식이 있다. ABR 서비스는 지연에 영향을 받지 않으며 손실에 민감한 트래픽을 그 응용으로 가정하였으며 망의 상태에 따라 지연의 변화가 심하다.

TCP<sup>[2]</sup> (Transmission Control Protocol) 는 가장

\* 삼성전자 정보통신 사업부

\*\* 한양대학교 전자통신전파공학과

논문번호 : 99136-0413, 접수일자 : 1999년 4월 13일

※ 본 연구는 1999년도 한양대학교 교내 연구비에 의하여 연구되었음.

널리 이용되는 데이터 전송 방식으로 윈도우 기반 흐름제어를 이용하여 신뢰할만한 데이터 전송을 한다. TCP는 IP (Internet Protocol) 상에서 동작하며, IP는 ATM상에서 동작하게 되므로 TCP가 ABR 서비스를 이용할 경우 두개의 독립적인 제어 알고리즘이 동작하게 된다.

TCP의 윈도우 기반 흐름제어는 수신 측에서 보내오는 ACK (ACKnowledgement) 신호를 이용하여 패킷의 손실을 예측하는 방식이다. 이러한 방식은 실제로 패킷의 손실이 발생해야 망의 혼잡 상태를 감지할 수 있으며, 때로는 잘못된 예측으로 불필요한 재전송을 하기도 한다.

ABR에서 사용하는 RM 셀 내부의 ER 필드는 망의 각 노드들이 수용 가능한 전송률 중 가장 작은 값을 저장하고 송신 측으로 되돌아오며, 송신 측에서는 이 전송률에 맞추어 정보를 전송해야한다. 그러나, TCP는 이러한 ABR의 전송률 기반 종단간 패 루프 제어 알고리즘과 무관하게 설계되어있기 때문에 TCP가 ABR 서비스 클래스를 통하여 전송되는 경우 두 알고리즘의 부조화에 의해 문제가 발생할 수 있다.

본 논문에서는 ABR 제어 알고리즘에 의하여 생성된 ER 값을 TCP 계층에 전달한다는 가정 하에 TCP의 윈도우 크기를 이 ER 값과 RTT (Round Trip Time) 을 이용하여 동적으로 결정함으로써 두 알고리즘 사이의 부조화를 해결하고 패킷 손실 이전에 망의 혼잡 상황에 대처함으로써 TCP over ABR의 성능을 향상시킬 수 있음을 보인다.

TCP over ABR의 성능 개선을 위한 수많은 연구가 수행되었거나 수행중이고 많은 개선방안들이 제안되었지만<sup>3),7)</sup> 현재까지 RM 셀 내부의 ER 값을 이용한 TCP 윈도우 크기 결정은 제안되지 않았다.

## II. TCP over ABR

### 1. TCP의 문제점

TCP는 패킷 망에서 훌륭한 성능을 보이는 프로토콜이지만 몇 가지 잘 알려진 단점들이 있다. 그중 가장 대표적인 것으로 폭주감지가 늦다는 것과 재전송에 의한 손실이 크다는 것을 들 수 있다. TCP에서의 폭주감지는 패킷 손실에 의해 이루어지며 패킷이 실제로 손실되어야만 망의 폭주상태를 감지할 수 있고, 필연적으로 재전송에 의한 손실을 감수해야만 한다. 또한, 공정성을 보장하지 못한다는 것 또한 TCP의 큰 단점중 하나이다.

### 2. TCP over ABR의 문제점

ABR에서는 RM 셀을 이용한 전송률 기반 종단간 패 루프 제어 알고리즘에 의해 송신 측의 송신률을 결정함으로써 셀 손실을 줄이고 스위치 버퍼 요구를 최소화한다. 이러한 ABR의 흐름제어는 망의 상황에 따라 송신 측에 적절한 송신률을 강요함으로써 안정적인 망의 상황을 유지하는 방식이다.

그러나, TCP는 이러한 알고리즘과는 무관하게 동작한다. TCP에서의 정보 전송은 대략 1 RTT 마다 윈도우 크기만큼의 정보가 전송되므로 '윈도우 크기 / RTT' 이라고 볼 수 있으나 이러한 전송률은 ABR 제어 알고리즘에 의하여 생성된 ER 값과 관련 없이 결정된다.

TCP가 ABR 상에서 동작할 경우, 이 두 값 (ER, 윈도우 크기) 의 부조화에 인하여 망에서 수용 가능한 만큼의 정보를 보내지 못하게되어 사용가능한 망 자원을 낭비하게 되거나, 망에서 수용 가능한 전송률 이상으로 정보를 보내게 되어 과도한 B-TTE (Broadband Terminal Equipment) 버퍼 요구를 보이게 된다.

## III. 수정된 TCP 흐름제어 알고리즘

### 1. 알고리즘

앞 절에서 밝힌 TCP over ABR의 문제점들을 해결하기 위하여 보다 신뢰할 수 있는 정보인 ATM 계층의 RM 셀 정보를 이용하여 TCP 윈도우 크기를 결정하는 방법을 제안한다. 본 알고리즘에서는 ATM 스위치 알고리즘으로 ERICA 스위치 알고리즘<sup>8)</sup>을 이용하였으며, 송신측 ATM 계층에서 망을 거쳐 전달받은 RM 셀 내의 ER 값을 TCP 계층으로 전달한다고 가정하였다.

본 알고리즘은 TCP-Reno를 기본으로 하고 있으며 송신측의 알고리즘만을 다음과 같이 일부 수정한다.

#### (1) 슬로우 스타트 메커니즘

처음 1 패킷을 전송한 후 바로 폭주 방지 상태가 되도록 한다. 1 패킷에 의해 RM 셀이 생성되어야 하므로 MSS는 1488(48\*31)bytes 이상으로 하여야 한다.

#### (2) 폭주 방지 메커니즘

수신측에서 보낸 ACK를 받을 때마다 다음의 식을 이용하여 새로운 윈도우 크기를 결정한다.

윈도우 크기 = last\_ER \* (48/53) \* (31/32)  
 \*(TCP MSS/TCP MSS + 56bytes)\* estimated\_  
 RTT \* safety\_factor

### (3) 재전송 메커니즘

재전송 메커니즘은 TCP-Reno의 재전송 메커니즘을 그대로 이용한다.

## 2. 분석

한 패킷이 전송되어 ACK가 돌아올 때까지 윈도우 크기의 변화가 없다면 TCP에서의 throughput은 '윈도우 크기 / RTT' 이라고 볼 수 있다. ABR에서 한 연결이 사용 가능한 대역폭인 ER 값에 ATM 셀의 오버헤드(48/53 = (ATM payload/ATM cell size)), RM 셀 오버헤드 (31/32), TCP, IP, AAL5, RFC 1577의 오버헤드들(56 bytes = 20 TCP header + 20 IP header + 8 RFC 1577 + 8 AAL5)을 위 식과 같이 제거하면 실제로 TCP 계층에서 전송 가능한 전송률이 구해진다. 여기에 estimated\_RTT를 곱하면 전송 가능한 양을 알 수 있다.

TCP에서 윈도우 크기는 이미 망에 내보내졌으나 ACK를 받지 못한 정보의 양( $W_1$ )을 포함한다. 따라서, ER 값을 이용하여 계산된 망이 수용 가능한 윈도우 크기( $W_2$ )에서  $W_1$ 을 뺀 정보의 양을 보낼 수 있게된다. 이때, 망 상황의 변화로 윈도우 크기가 줄어들어  $W_2 < W_1$ 인 경우에는 정보를 보낼 수 없다.

구해진 값을 직접 폭주 윈도우 크기로 사용하면 윈도우 크기가 증가할 때는 원하는 크기로의 증가가 가능하지만 윈도우 크기를 감소시켜야 하는 경우에는 ACK를 받지 못하고 이미 전송된 양( $W_1$ ) 때문에 제한된 감소만이 가능하다. 다시 말해서  $W_2$ 가  $W_1$ 보다 작은 경우,  $W_2$ 로 윈도우 크기를 감소시켜야 하지만 이미 망에 내보내진 양,  $W_1$  때문에 윈도우 크기는 최소  $W_1$ 이 된다. 따라서 구해진 값에 적절한 safety\_factor( $S < 1$ )를 곱하여 사용함으로써 TCP 계층의 전송률을 조절한다.

또한, safety\_factor를 이용함으로써 망 상황의 급격한 변화와 부정확한 RTT의 영향을 보상하는 효과도 기대할 수 있다.

이상과 같은 알고리즘의 적용으로 앞 절에서 언급된 문제들을 다음과 같이 해결하고 전체적인 TCP의 성능 향상을 가져올 수 있다.

### (1) TCP 자체의 문제점

기존의 TCP가 망 상황에 무관하게 정보를 전송하고 패킷의 손실로 망의 혼잡을 예측하던 것에 반

해, 제안된 방식은 망에서 생성된 ER 값을 이용하여 망의 상황을 알아내고 이에 맞추어 적정량의 정보를 전송함으로써 망의 혼잡을 사전에 예방하고 그에 따르는 재전송도 줄일 수 있다. 또한, ER 값 자체가 각 송신자들의 공정성을 고려한 값이기 때문에 ER 값에 따른 TCP 전송률 조절은 TCP의 공정성도 증가시킬 수 있다.

### (2) TCP over ABR의 문제점

제안된 방식의 사용은 ABR 흐름제어에 의해 송신 측에 강요되는 송신률에 맞게 TCP의 윈도우 크기를 조절함으로써 B-TE에서 망으로 전송할 수 있는 전송률(ER)과 응용계층에서 B-TE로 유입되는 정보의 전송률(윈도우 크기/RTT)을 맞추어줌으로써 B-TE의 버퍼 요구를 감소시켜준다. 또한 TCP 패킷이 BTE 버퍼에서 머무는 시간을 줄여주어 TCP의 전송 지연을 줄여준다.

## IV. 시뮬레이션

본 시뮬레이션은 CLASS<sup>[9]</sup> (Cell Level ATM Service Simulator) 를 이용하여 수행되었으며, 시뮬레이션을 위하여 III.1절의 내용으로 일부 수정되었다.

### 1. 시뮬레이션 모델

시뮬레이션은 아래 그림과 같이 ABR 서비스 상에서 동작하는 5명의 FTP 사용자와 n명의 on-off CBR 사용자가 150Mbps의 링크를 공유하고 있는 경우에 대해 행해졌으며, 각 링크의 길이  $l_1, l_2, l_3$ 는 LAN 및 WAN 환경을 가정하기 위하여 각각 1Km, 100Km, 1Km 와 1Km, 1Km, 1Km의 길이에 대하여 수행되었다.

Estimated\_RTT의 계산은 기존의 방법을 이용하였으며, 무한 소스 모델을 가정하였다.

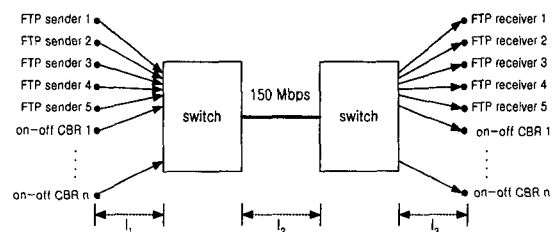


그림 1. 시뮬레이션 모델

본 시뮬레이션에 사용된 파라미터들은 다음과 같다.

- 링크 용량 = 150 Mbps
- 링크 버퍼 = 1,000 cells (on-off CBR 사용자용)  
2,500 cells (FTP 사용자용)
- on-off CBR 소스 on time = 84 ms  
off time = 84 ms
- TCP MSS(Maximum Segment Size) = 9,140 bytes
- TCP timer granularity = 50ms
- TCP maximum 윈도우 크기 = 20 segments  
= 182,800 bytes
- 사용자 버퍼 크기 = 1,000,000 cells
- ABR Nrm = 32
- PCR = 150 Mbps
- MCR = 0.1 Mbps
- ICR = 5 Mbps
- RIF = 1
- RDF = 1
- TBE = 512 cells

## 2. 결과 및 분석

TCP의 성능 파라미터로 throughput (Mbps)과 재전송 비율 (%)을 구하였다. Throughput은 전송된 양을 전송된 시간으로 나누어 구하였으며 재전송

비율은 전체 전송된 패킷의 수로 재전송된 패킷의 수를 나누어 계산하였다.

Background 트래픽은 다음 표 1, 2에서와 같이 고정된 대역폭을 TCP만 이용하는 경우에서의 성능을 측정하기 위하여 background 트래픽이 없는 경우를 가정하였고, 망의 버스티니스가 큰 경우와 적은 경우를 위하여 각각 1명의 100 Mbps on-off CBR 사용자와 10명의 10 Mbps 사용자를 가정하였다. 또한 5명의 10 Mbps 사용자는 보다 망 자원의 여유가 많이 있는 경우에서 제안된 알고리즘의 동작을 알아보기 위하여 가정하였다.

각각의 망 상황에서 safety\_factor (S)를 변화시켜 가며 시뮬레이션한 결과는 다음 표 1, 2와 같다.

앞의 표 1, 2에서 볼 수 있듯이 모든 경우에서 제안된 방식을 이용하여 성능의 향상을 가져올 수 있음을 볼 수 있다. 그림 2, 3에서와 같이 TCP-Reno에서는 망의 상황에 상관없이 무조건적인 윈도우 크기의 증가를 보이다가 정보의 손실과 그에 따르는 재전송 및 윈도우 크기의 증가를 반복하고 있으며, 변경된 알고리즘을 적용한 경우에는 망의 상황에 따라 윈도우 크기를 유동적으로 변경함을 알 수 있다.

예상되는 바와 같이 큰 safety\_factor에서는 큰 증가와 한정된 감소로 인해 TCP 성능의 향상이 감소

표 1. 시뮬레이션 결과 (WAN)

Background Traffic	Performance Parameter	변경 전	변경 후					
			S = 0.2	S = 0.3	S = 0.4	S = 0.5	S = 0.6	S = 0.7
0 Mbps	Throughput(Mbps)	92.6	126.0	126.0	128.8	128.8	129.2	121.8
	재전송 비율(%)	14.4	2.1	2.1	0.8	0.5	0.5	3.7
100 Mbps * 1	Throughput(Mbps)	67.4	82.3	82.7	83.3	81.6	78.7	75.5
	재전송 비율(%)	9.7	4.5	1.0	2.2	3.4	6.2	7.3
10 Mbps * 10	Throughput(Mbps)	65.6	85.5	86.0	83.9	82.0	77.5	74.1
	재전송 비율(%)	10.9	1.7	1.0	2.2	3.4	6.2	8.4
10 Mbps * 5	Throughput(Mbps)	75.6	105.9	107.0	107.6	105.8	100.5	94.4
	재전송 비율(%)	15.0	1.8	1.3	0.8	1.7	4.4	7.4

표 2. 시뮬레이션 결과 (LAN)

Background Traffic	Performance Parameter	변경 전	변경 후					
			S = 0.2	S = 0.3	S = 0.4	S = 0.5	S = 0.6	S = 0.7
0 Mbps	Throughput(Mbps)	94.2	129.4	128.1	129.5	129.1	127.0	120.2
	재전송 비율(%)	14.0	0.9	1.8	0.7	0.8	1.6	4.4
100 Mbps * 1	Throughput(Mbps)	67.4	82.6	82.4	81.7	79.6	76.6	73.9
	재전송 비율(%)	10.3	4.6	4.3	4.2	5.2	6.7	8.1
10 Mbps * 10	Throughput(Mbps)	65.8	85.4	85.8	83.1	81.3	77.7	74.3
	재전송 비율(%)	10.6	1.7	1.0	2.7	3.8	6.2	8.4
10 Mbps * 5	Throughput(Mbps)	76.6	106.8	107.3	107.4	105.1	101.0	95.3
	재전송 비율(%)	14.8	1.8	1.3	0.9	2.0	4.1	7.0

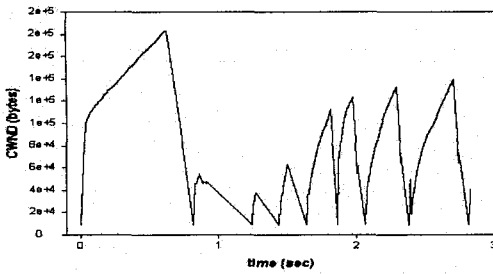


그림 2. TCP-Reno에서의 윈도우 크기 변화  
(background traffic : 10Mbps \* 10 )

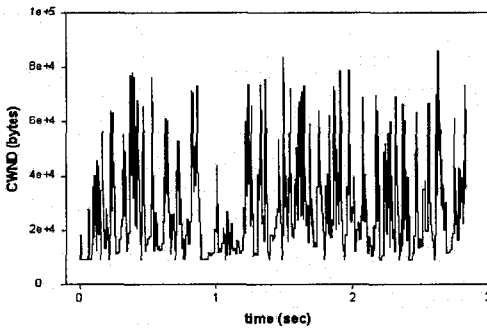


그림 3. 제안된 알고리즘에서의 윈도우 크기 변화  
(background traffic : 10Mbps \* 10 )

하고, 작은 safety\_factor에서는 작은 증가와 작은 감소로 ER 값에 적절히 대응하지 못함으로써 성능이 감소하였다.

TCP의 성능은 safety\_factor가 0.4 부근일 때 LAN 및 WAN 환경 모두에서 throughput의 현저한 향상과 재전송 비율의 감소를 가져올 수 있다. 위와 같은 결과는 throughput의 향상 자체에도 큰 의미가 있지만, 재전송 비율을 줄임으로써 불필요한 망 자원의 낭비를 줄였다는 점에서도 그 의미를 찾을 수 있다.

제안된 알고리즘은 TCP 흐름제어의 상당부분을 ABR 흐름제어에 의존하는 방식이다. 따라서 망 상황에 따른 TCP의 능동적인 대처가 가능해진다. 또한, fairness와 stability등의 경우도 ERICA 스위치가 제공하는 ER 값을 사용하게 됨으로써 자연스럽게 상속받을 수 있다.

본 시뮬레이션은 무한 소스와 충분한 B-TE 버퍼를 가정하고 실행되었다. 무한 소스가 아닌 경우와 B-TE 버퍼에서 오버플로우가 발생하는 경우 등에 대한 고려와 연구가 이어질 것이다.

## V. 결론

본 논문에서는 ATM 계층의 RM 셀 정보를 상위 계층인 TCP 계층으로 전달하여 TCP over ABR의 성능을 개선하는 알고리즘을 제안하였다.

제안된 알고리즘은 시뮬레이션 결과와 같이 LAN 및 WAN 환경과 다양한 망의 background 트래픽에 대해서 현저한 성능의 향상을 보였다. Safety\_factor가 0.4일 때 throughput은 각 망의 상황에 따라 최고 42.3%에서 21.2%까지의 증가를 보였고 재전송율도 현저한 감소를 보였다.

## 참고 문헌

- [1] ATM-Forum, "ATM-Forum Traffic Management Specification Version 4.0," Apr. 1996.
- [2] V. Jacobson, "Congestion Avoidance and Control," *Proceedings of the SIGCOMM'88 Symposium*, pp.314-32, Aug. 1988.
- [3] Shiv Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, Fang Lu, Saragur Srinidhi, "Performance of TCP/IP over ABR Service on ATM Networks," *Globecom '96*, Nov. 1996.
- [4] Douglas E. Comer, John C. Lin, "TCP Buffering And Performance Over An ATM Network," *Journal of Internetworking: Research and Experience*, Vol.6(1), pp 1-13, Mar. 1995.
- [5] Pil-Joong Kim, Woon-Sik Kim, Sang-Ha Kim, Pyung-Dong Cho, "TCP Flow Control Mechanism for ATM Networks," *ICICS '97*, Sep. 1997.
- [6] Per Jahanesson, Elin Wedlund, Johan M. Karlsson, "Interaction Between TCP Flow Control and ABR Rate Control," *Proceedings of the IEEE ATM'97 workshop*, May 1997.
- [7] Kai-Yeung Siu, Hong-Yi Tzeng, "Performance of TCP over ATM with time-varying available bandwidth," *Computer Communications*, Vol.19, pp.927-936, Sep. 1996.
- [8] Raj Jain, Shiv Kalyanaraman, Rohit Goyal, Sonia Fahmy, and Ram Viswanathan, "ERICA Switch Algorithm: A Complete Description," *AF-TM 96-1172*, Aug. 1996.
- [9] CSELT (Centro Studi E Laboratori Tele-

2000년 1월~현재 : 삼성전자 정보통신 사업부  
 <주관심 분야> 초고속정보통신, 이동통신시스템

한국통신학회 논문지 제23권 제3호 참조