

ABR과 UBR 서비스 상에서의 인터넷 프로토콜: 문제점, 해결방안, 그리고 성능평가

박 승 섭[†] · 육 동 철^{††}

요 약

초고속 인터넷에서 멀티미디어 트래픽 증가로 인해 ATM 네트워크는 인터넷의 많은 부분에 걸쳐서 백본 네트워크의 역할을 하고 있다. 본 논문은 ATM 서비스의 ABR과 UBR상에서 TCP/IP 성능을 향상시키기 위해, 시뮬레이션 기법을 통해 높은 처리율과 보다 나은 공정성을 나타낸 연구이다. 비록 TCP제어는 트랜스포트 계층에서 이루어지지만, TCP가 ABR/UBR 서비스를 사용할 때에는 ATM층의 여러 가지 방법, 즉 EPD, PPD, RED, EFCI, ER 등에 의해 제어된다. 따라서 ATM층에서 한 개의 셀이 손실되면, 한 개의 TCP 패킷의 손실로 이어지고, 더불어 스위치의 VC 수의 증가로 인해서도, 처리율과 공정성이 낮아진다. 이러한 문제점을 개선하기 위해서, 본 연구에서는 ABR서비스에서는 EFCI와 ER의 효율적인 파라미터 제어 운용 방안을 제안하고, UBR 서비스에서는 버퍼관리 방법을 제시하여, 시뮬레이션 결과를 통해 개선된 처리율과 공정성을 나타내었다.

Internet Protocols Over ABR and UBR Services : Problems, Approaches, and Their Evaluation

Seung-Seob Park[†] · Dong-Cheol Yuk^{††}

ABSTRACT

As the proliferation of multimedia traffic over High-speed Internet increases, ATM network will be vital to adopt as backbone network over various parts of Internet. In this paper, we investigate the performance of TCP/IP traffic flow over ABR and UBR of ATM service to study for the high throughput and good fairness by simulation technique. Although TCP is run in the transport layer, it is controlled by several methods, e.g, EPD, PPD, RED, EFCI, ER etc., in ATM layer when TCP uses the ABR/UBR service. Therefore, if one cell is discarded in ATM layer, a packet of TCP will be lost. And, also, along with the increasing of the number of VC among switches, the throughput and fairness will be degraded. In order to improve these degradations, we propose the effective parameter control operations of EFCI and ER on ABR service, and also suggest the buffer management methods on UBR service. Finally, through the simulation results, the improved throughput and fairness are shown.

1. 서 론

ATM(Asynchronous Transfer Mode)은 멀티미디어

통신을 가능하게 하는 고속 광대역 통신망(B-ISDN)을 실현하기 위한 기본 방식으로 탄생된 것으로 멀티미디어 서비스에 다양한 QoS를 제공해야 한다. 따라서 ATM 포럼에서는 ATM서비스 범주를 CBR(Constant Bit Rate), rt-VBR(real time Variable Bit Rate), nrt-VBR(non-real time Variable Bit Rate), ABR(Available Bit

[†] 정 회 원 : 부경대학교 컴퓨터멀티미디어공학부 교수

^{††} 준 회 원 : 부경대학교 대학원 전자계산학과

논문접수 : 1999년 9월 19일, 심사완료 : 1999년 11월 10일

Rate), UBR(Unspecified Bit Rate)의 5가지 서비스 범주로 표준화하였다[1]. 인터넷의 TCP(Transport Control Protocol)는 연결 지향 프로토콜로서 현재 인터넷 데이터 전송을 위한 프로토콜로 가장 많이 사용되고 있다. ATM 상에서 TCP 데이터는 신뢰성 있게 전송이 보장되어야 한다. 다시 말해서 데이터 전송을 위한 ATM 서비스 범주 ABR과 UBR상에서 TCP는 안정된 품질로 전송되어야 한다[8]. 그러나 트랜스포트 계층에서 윈도우 방식을 이용하여 폭주 제어를 하는 TCP와 ATM의 각 서비스 범주에서 하는 폭주 제어는 서로 다른 방식으로 수행한다.

TCP는 데이터를 전송하기 위해 윈도우 방식을 이용하며 slow start, 혼잡 회피, 재전송 단계로 구성된 폭주제어와 오류제어를 수행한다[2]. UBR 서비스는 위에서 언급한 서비스 범주 중에서 가장 경제적인 반면에 우선 순위가 가장 낮은 서비스이며, UBR은 별도의 폭주 제어가 없어서 망에서 셀 손실 가능성도 높다. 이러한 단점을 보완한 셀 폐기 정책인 EPD(Early Packet Discard)나 PPD(Partial Packet Discard)[6], RED(Random Early Detection)[9] 등을 사용해 망에서 불필요한 대역 낭비를 줄이는 알고리즘을 함께 사용하기도 한다.

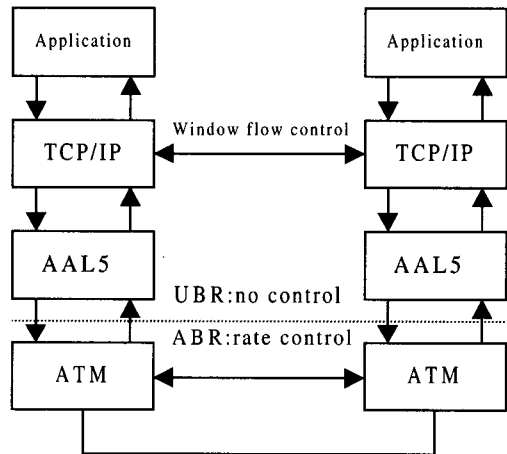
ABR에서는 이진 피드백 방식(EFCD)과 명시적율(ER)스위치 방식을 사용하여 데이터 셀의 EFCI 비트(RM(Resource Management) 셀의 정보를 통해 ACR(Allowed Cell Rate)를 조정한다. 이렇게 TCP 데이터 서비스와 ATM의 ABR과 UBR 서비스 범주에서 혼잡 제어 방식과 오류 제어 방식이 서로 다르게 처리하기 때문에 분해, 조립에서의 지연, 재전송 및 세그먼트의 순서가 맞지 않는 경우가 발생하여 여러 가지 문제가 야기된다[11].

본 논문 구성은 서론에 이어, 2장에서는 TCP 혼잡 제어 방식과 TCP가 UBR과 ABR에서 동작하는 방식을 설명하였으며, 3장에서는 시뮬레이션 환경을 나타내었고, 4장에서 시뮬레이션 결과 분석을 통해 문제점을 밝히고, 5장은 문제점에 대한 개선 방안을 제시하였으며, 마지막으로 6장에서는 결론에 대해 서술하였다.

2. 트랜스포트 계층과 ATM 계층에서의 혼잡 제어 방식

네트워크에서의 응용프로그램 데이터는 TCP/IP계층으로 내려가 세그먼트로 나누어지면서 TCP헤더와 IP

헤더가 추가된다. 그리고 AAL 계층에 가서는 상위 계층의 세그먼트들이 캡슐화되고, ATM 계층에 가서는 ATM헤더가 추가되어진다. 그리고 각 계층에 따라 혼잡제어는 다른 방식으로 처리된다. (그림 1)에서는 이러한 프로토콜 스택을 보여주고 있다. 다음 절에서는 트랜스포트 계층과 ATM 계층에서의 혼잡제어 방식을 설명한다.



(그림 1) TCP/IP와 ATM 프로토콜 스택

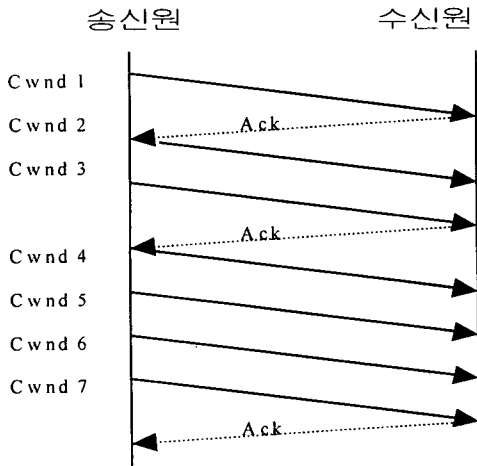
2.1 TCP 혼잡 제어 방식

TCP는 인터넷에서 가장 많이 사용되고 있는 데이터 전송을 위한 프로토콜이다. 데이터를 신뢰성 있게 전송하기 위해서 종단간 흐름 윈도우 방식을 사용한다. 그리고 TCP의 혼잡제어로서는 slow start, 혼잡 회피(congestion avoid), 빠른 재전송(fast retransmission) 등이 있다.

Slow start는 송신원에서 수신원에 연결 요청 수락 후, 초기 데이터를 전송하고 수신측 TCP 컨넥션은 송신원에 ACK를 보내서 수신 성공을 알린다. 이때 송신원은 RTT(Round Trip Time)를 계산하고, RTT를 기본으로 RTO(Retransmission Time Out)를 계산한다. 타이머가 만료 될 때까지 ACK 신호를 받지 않으면 세그먼트 손실로 간주하고 전송한 세그먼트부터 다시 전송하는 재전송 방법을 사용한다. 또한 에러가 발생되지 않을 경우에는 최대 윈도우 크기만큼의 세그먼트를 ACK 없이 송신할 수 있다.

(그림 2)와 같이 Slow start 동안에는 매 ACK에 대해서 폭주 윈도우(Congestion window : Cwnd)의 크기

를 하나씩 증가시키며, 폭주 윈도우의 크기 증가는 ss-threshold (slow start threshold)에 이르거나 세그먼트의 손실이 감지될 때까지 계속된다.



(그림 2) Slow start 제어와 ACK 신호

이러한 방식으로 망에서 윈도우의 크기를 동적으로 조정하여 혼잡을 회피하는데, 혼잡 시나 전송한 세그먼트에 대한 타이머가 만료되면 현재 윈도우 크기의 반을 ss-threshold로 설정하고 폭주 윈도우의 크기를 1로 설정한 후 다시 slow start 단계로 들어가 손실된 세그먼트부터 재전송하는 방법으로 폭주윈도우의 크기가 ss-threshold에 이르거나 세그먼트의 손실이 감지될 때까지 계속 증가하게 된다. 또한 폭주윈도우의 크기가 ss-threshold와 같아지거나 커지면 혼잡회피 단계로 들어가는데, 현재의 폭주 윈도우의 크기를 a 로 가정하면 매 ACK마다 $1/a$ 만큼의 폭주 윈도우가 증가하게 된다. 이는 폭주 윈도우의 크기가 선형적으로 증가하며, 최대 윈도우크기에 도달하거나 세그먼트의 손실이 감지될 때까지 계속된다. 빠른 재전송은 수신원에 순차번호에 맞지 않는 세그먼트가 수신되면 잘못 수신된 세그먼트 순차번호를 ACK에 실어 송신하여 올바른 순차번호를 갖는 세그먼트가 수신될 때까지 계속 보낸다. 그러면 송신원은 중복된 ACK가 n 개 이상 수신되면 손실로 간주하여 재전송한다. 이것은 재전송 타이머가 만료될 때까지 기다려야 하는 시간을 줄이는 방법이다[7].

2.2 ATM층에서의 폭주제어

2.2.1 UBR과 UBR+EPD상에서의 동작

UBR 서비스는 여러 가지 서비스에서 가장 경제적인 반면에 우선 순위가 가장 낮은 서비스이며, CBR, VBR, ABR 서비스가 사용하고 남은 대역폭을 사용하는 서비스이다. 따라서 송신원에서 PCR(Peak Cell Rate)로 셀을 전송할 때 UBR은 별도의 흐름제어는 없다. 이 제어를 위해 트랜스포트 계층의 TCP 제어방식에 의존한다. 만약 ATM 스위치의 버퍼가 제한적일 때, ATM 계층에서의 하나의 셀 손실은 트랜스포트 계층에서 하나의 패킷 손실로 이어진다. 이러한 현상으로 낮은 결과와 VC(virtual channel)수 간의 불공정한 결과를 초래할 수 있고, 망에서 버퍼길이에만 의존하기 때문에 패킷이 손실 될 가능성도 가장 높으며, QoS도 보장이 확실하지 않다.

이러한 문제를 해결하기 위해서 UBR 서비스에서는 PPD나 EPD, RED 알고리즘 등을 사용해서 망에 불필요한 데이터가 전송되는 것을 억제하여 대역폭 낭비를 막는다. EPD는 버퍼에 임계치를 두어 이 임계치를 넘는 패킷의 첫 번째 셀이 있으면 그 첫 번째 셀과 같은 가상채널 식별자를 가지는 패킷 전부를 폐기하는 방법으로 버퍼 오버플로우로 셀 손실로 인한 패킷의 손실을 방지하여 불필요한 재전송과 대역폭 낭비를 막기 위해서이다. 보다 상세한 내용은 참고문헌[12]에서 보여진다. 따라서 본 연구는 UBR 서비스에서 좋은 성능을 가지는 EPD 알고리즘을 사용하였다.

2.2.2 ABR상에서의 동작

UBR과는 반대로 ABR 서비스에서는 ATM 계층 자체에 율 기반 흐름제어를 제공한다. 그리고 스위치 폭주 제어와 모든 VC간에 공정하게 자원을 할당하기 위해 RM 셀의 피드백 정보에 따라 ACR를 조정할 수 있는 ER(Explicit Rate)방식과 데이터 셀의 EFCI 비트로 폭주여부를 확인하는 EFCI(Explicit Forward Congestion Indication)방식이 있다. 가장 간단한 스위치 방식으로 스위치에 있는 모든 VC는 공동의 FIFO큐를 공유하고 큐의 길이가 감시된다. 만약 큐의 길이가 폭주 발생이라고 설정된 임계치를 넘으면 그 스위치를 지나는 데이터 셀들의 헤더에 EFCI 비트를 설정한다. 피드백 되어온 정보에 의해 송신원은 ACR을 MCR

(Minimum Cell Rate)까지 줄이고, 만약 큐의 길이가 임계치 이하로 떨어지면 폭주를 해지하고 스위치를 지나는 셀들은 EFCI 비트를 설정하지 않고, 다시 전송량을 늘리는 방식으로 송신원은 피드백하는 데이터 셀 헤더에 있는 EFCI 비트 설정여부에 따라 송신 전송대역폭을 조절한다. ER방식은 EFCI제어 방식에 비해서 더 빠르고, beat down 문제를 해결할 수 있다. 송신원은 데이터 셀과 함께 RM 셀을 수신원에 전송하는데, 여러 스위치를 통과한 데이터 셀과 RM 셀 필드 정보에 의해 폭주 정도 여부와 공정한 대역폭을 수신원은 계산해서 송신원에 피드백 시킨다. 이에 송신원은 피드백되어 온 RM 셀의 각 CI, NI, ER, CCR 등의 필드 값을 이용해서 ACR를 조정한다. 이 ACR 조정은 (그림 3)과 같다.

```

if CI bit in RM cell = 0 // 비폭주시
  if ER > ACR then
    ACR = ACR + RIF * PCR //ACR증가
  if ER < ACR then
    ACR = ER
if CI bit in RM cell = 1 // 폭주시
  ACR = ACR - ACR * RDF //ACR감소
  if ER < ACR then
    ACR = ER
    
```

(그림 3) ER 스위치 방식의 의사 코드

이상과 같이, 각 계층의 제어 방식을 설명하였으며, ATM Forum에서 규정한 Traffic Management Version 4.0(TM 4.0)에서 지정한 EFCI 방식과 ERICA(Explicit Rate Indication Congestion Avoidance)방식[5]을 사용해서 성능 평가를 하였다.

3. 시뮬레이션 환경

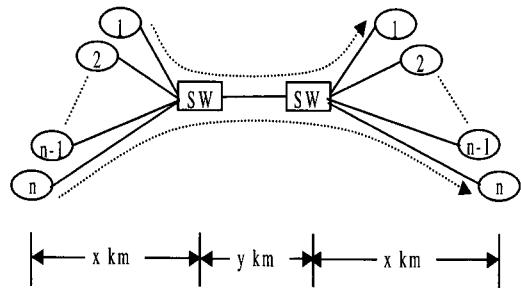
본 장은 시뮬레이션 망 모델과 파라미터에 대해 설명한다. 시뮬레이션 도구로써는 C++로 작성되어진 YATS ATM 시뮬레이션 도구[3]를 사용해서 시뮬레이션 하였다.

3.1 시뮬레이션 모델

UBR과 ABR 서비스 상에서 TCP의 성능 분석을 위해서 (그림 4)와 같이 하나의 병목 구간을 가지는 peer to peer 모델로 설정하였다. 전송방향은 단방향트래픽

(unidirectional traffic)으로만 데이터를 전송하게 한 모델이며, WAN이나 LAN 환경으로 구분하지 않고 병목 구간의 링크 속도는 150Mbps로 거리는 300Km로 가정하였다. TCP 컨넥션의 수는 가변적으로 최대 n만큼의 수를 가지고, 그리고 각 TCP 컨넥션과 스위치 사이의 전송거리는 $n * 0.56$ km로 설정하였다.

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



(그림 4) 시뮬레이션 망 모델

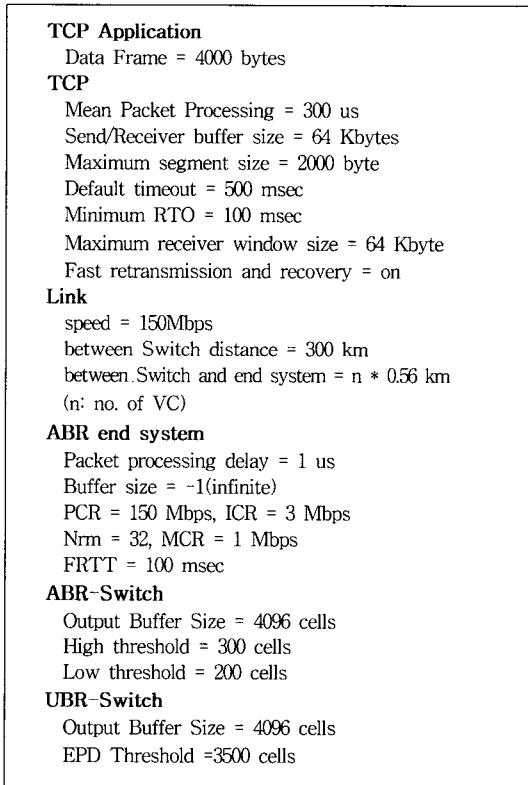
그리고 송신원의 TCP 데이터는 고정 프레임 길이 만큼 지속적으로 동시에 전송하게 하였다.

3.2 시뮬레이션 파라미터 설정

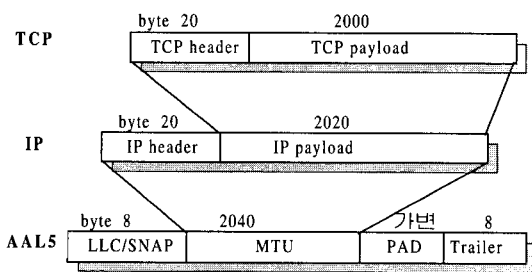
각 UBR, ABR, TCP 컨넥션, 스위치의 기본 파라미터 설정은 (그림 5)와 같이 나타내었다.

정의된 파라미터 값으로써 성능 개선에 대해서 다룰 때에는 버퍼크기, EPD 임계치, RDF(Rate Decrease Factor), RIF(Rate Increase Factor), 스위치 파라미터 AI(Averaging Interval)를 변경하였다. 이 이유는 임의적으로 폭주 상태를 만들어 시뮬레이션하기 위해 설정한 값이다. RIF와 RDF는 ATM Forum에서 ABR서비스 트래픽을 위한 파라미터로 2의 승수 만큼 증감할 수 있도록 정의되어 있다.

그리고 ATM 계층에서는 TCP 최대 세그먼트 크기 2000 바이트에 TCP 헤더 20바이트와 IP 헤더 20바이트, LLC 헤더 8 바이트, AAL 트레일러 8 바이트에 패딩(padding)과 ATM 층의 헤더 5바이트가 합쳐져서 43개의 셀이 된다. 그래서 최대 이용 가능한 처리율은 $2000 \div 2279 = 0.87$ 로 130.5 Mbps가 된다. (그림 6)은



(그림 5) 시뮬레이션 파라미터 설정값



(그림 6) AAL5 상에서의 TCP/IP 프로토콜 포맷

AAL 계층상에서의 TCP/IP의 프로토콜 포맷을 보여주고 있다.

4. 시뮬레이션 결과 분석

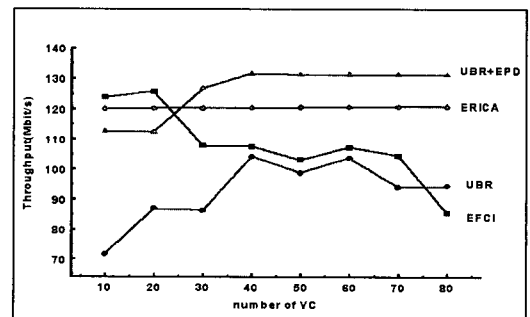
성능 평가 요소로서는 각 VC 수에 따른 처리율과 공정성(fairness)을 기준으로 하였으며 거리에 따른 처리율과 공정성은 평가하지 않았다. 평가 요소인 공정

성은 서로 다른 VC 간 대역폭 할당에 있어서 공평하게 할당받았는가를 평가하는 기준으로 다음 식과 같이 주어진다[4, 10].

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

여기서, $x_i = T_i / O_i$ 로 $T_i(i=1...n)$ 는 각 VC마다 성공적으로 도착된 패킷의 측정된 처리값으로 단위시간당 전송받은 양을 바이트로 계산하였고, $O_i(i=1...n)$ 는 최대-최소의 최적 처리율이고, n 은 전체 VC의 수이다. 여기에서 공정성의 기준값을 1로 설정하였는데, 공정성의 의미는 각 TCP 컨넥션 간의 대역폭을 공평하게 할당받았는가를 평가하는데, 예를 들어, 공정성이 0.4라고 가정한다면 대역폭을 많이 사용한 VC가 있는 반면에 대역폭을 낮게 또는 할당받지 못해서 기아 현상이 발생된 VC가 있다는 의미이다.

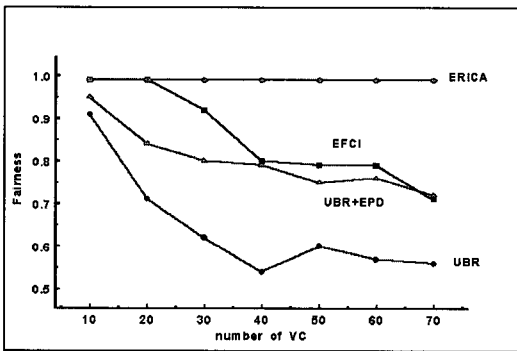
(그림 7)은 VC 수의 변화에 따른 처리율 변화를 보여 주고 있다. (그림 7)에서 나타난 것과 같이 UBR+EPD에서의 TCP가 가장 높은 처리율을 보여주고 있다. 반면에 UBR은 스위치의 버퍼 크기와 TCP혼잡 제어 방식에 의존하기 때문에 폭주시 망에 잘 대처하는 처리율을 나타내지 못하고 있다.



(그림 7) VC의 수에 따른 처리율

ERICA는 매우 안정적인 처리율을 보여주고 있는데, ERICA와 EFCI는 피드백 되어온 정보에 의해 전송률을 조정한다. 특히 ERICA와 같은 경우는 RM셀 필드의 ER값에 의한 각 대역폭을 할당받기 때문에 각 TCP 컨넥션이 전송할 수 있는 대역폭이 정해진다. EFCI같은 경우는 VC의 증가에 따라 더 빈번한 폭주 정보에 의해서 각 TCP컨넥션은 사용할 수 있는 대역

폭에 비해 전송율을 높일 기회를 자주 잃게 된다. 그러므로 VC가 많을 경우 처리율이 낮아진다. 그러나 (그림 7)에 대한 평가로써 처리율이 높다해서 각 VC는 충분히 자기 전송량 만큼 전송을 했다고 볼 수는 없다.



(그림 8) VC의 수에 따른 공정성

(그림 8)은 VC의 변화에 따른 공정성 평가를 나타내고 있는데 공정성 평가에서 ERICA가 각 VC에 대역폭을 공정하게 할당했음을 알 수 있다. 이는 VC당 RM셀 필드에 있는 ER값 계산에 의한 것으로 이는 TCP 데이터가 트랜스포트 계층에서 제어 받기보다는 ABR의 ER 방식의 율 흐름 제어(rate-base control)에 의해 조정됨을 알 수 있다. 그 외의 결과는 VC 증가에 따라 공정성이 감소되었고, 특히 UBR은 매우 낮은 결과가 나왔지만 EFCI 같은 경우는 VC가 더 증가하면 처리율 면이나 공정성 면에서는 UBR 보다 더 낮은 결과를 가져왔다.

그 이유는 UBR은 버퍼 크기의 오버플로우와 TCP 흐름 제어방식에 따르지만, EFCI는 ABR을 흐름제어에 의해서 전송량을 할당받는데 많은 송신원의 TCP 컨넥션이 동시에 스위치에 접근할 때, 한번의 폭주 발생 신호 후 끊임없는 폭주 발생 신호에 의해 ACR를 증가시킬 기회가 없기 때문이다. 그리고 UBR+ EPD는 높은 처리율에 비해 낮은 공정성을 보여주고 있다.

<표 1>은 데이터 VC수의 증가에 따른 손실 바이트와 재전송 바이트를 나타내었다. EPD를 사용했을 때에는 UBR보다 손실 바이트는 크게 나왔으나 재전송 바이트는 작게 나왔고, EFCI는 손실바이트에 비해 재전송 바이트가 크게 나왔다.

<표 1> VC의 수에 따른 손실과 재전송

VC 수	UBR		UBR+EPD		EFCI		ERICA	
	Lbytes	Rbytes	Lbytes	Rbytes	Lbytes	Rbytes	Lbytes	Rbytes
10	4578	9800	6141	9400	0	0	0	0
20	8750	9900	11521	9900	0	0	0	0
30	11088	11133	16227	9866	1627	3000	0	0
40	13752	10750	20842	10350	2581	4800	0	0
50	16022	10800	23310	10120	7470	7520	0	0
60	17966	10566	26640	10233	5086	4833	0	0
70	19789	11000	32272	10171	9665	9057	0	0

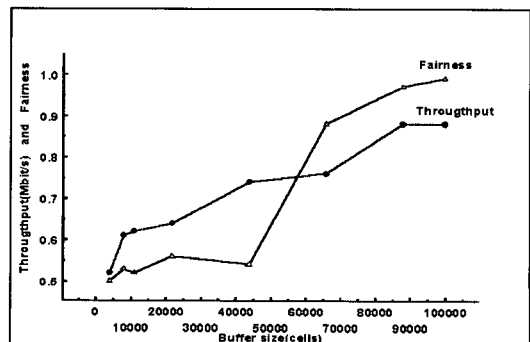
Note : Lbytes ; lost byte, Rbytes ; retransmission byte

5. 개선 방안

본 장에서는 4장의 결과 분석에 나타난 문제점과 비교 분석 결과를 가지고 각 문제점에 버퍼 관리와 파라미터의 변화에 따른 TCP 성능 개선 방안 대해서 기술하며, UBR과 UBR+EPD는 TCP의 윈도우크기나 세그먼트의 변화 등 TCP의 여러 파라미터에 의해서도 영향을 받지만, 본 연구에서는 TCP의 파라미터 변화를 배제한 버퍼 관리와 EFCI 및 ER은 ATM 계층의 파라미터의 효율적인 제어에 따른 개선을 도모하였고, 모든 VC 변화 상태에서가 아니라 처리율과 공정성이 최악인 VC의 수가 70인 경우를 모델 설정하여 개선방안 대해서 나타내었다.

5.1 UBR과 UBR+EPD상에서의 개선 방안

UBR과 UBR+EPD는 버퍼의 크기와 EPD 임계치에 영향을 받는다. 특히 UBR 그 자체는 버퍼의 크기에 의존하기 때문이다. ATM 계층에서 버퍼 오버플로우는 셀의 손실을 가져오고 트랜스포트 계층에서는 전



(그림 9) 버퍼 크기 변화에 따른 처리율과 공정성

제 패킷에 영향을 미칠 뿐 만 아니라 불필요한 패킷의 전송과 그에 따른 재전송으로 대역폭 낭비를 가져온다. 그러므로 버퍼의 변화에 따른 UBR 성능 개선에 대해서 (그림 9)와 같이 나타내고 있다. 위의 (그림 9)에서 보는 바와 같이 버퍼의 크기가 커질수록 처리율이 점차 증가됨을 알 수 있다. 그러나 공정성은 버퍼 용량이 충분할 때 급격히 증가하는데, 결론적으로 송신원의 TCP 컨넥션의 최대 윈도우 크기 64Kbyte×VC수 만큼 버퍼용량을 필요로 하고 있다.

UBR+EPD방식은 UBR방식의 단점을 보완하기 위한 셀 폐기(drop)정책으로 버퍼 오버플로우로 셀이 손실되어 불필요한 패킷이 되지 않게 하고 쓸모 없는 셀이 대역폭을 사용하지 않게 함으로써 대역폭 이용율을 증가시킨다. 셀 폐기는 EPD 임계치 제어에 따라 달라지는데 이것은 처리율과 공정성에 영향을 미친다.

VC의 수가 70이고 EPD 임계치가 3500셀일 때 버퍼 변화에 따른 처리율과 공정성을 <표 2>에 나타내었다. 나타난 결과는 고정된 임계치에서 버퍼의 크기가 증가를 해도 처리율과 공정성에는 아무런 변화가 없다는 것을 알 수 있다.

<표 2> 고정된 임계치와 버퍼크기 변화에 따른 처리율과 공정성

Buffer size (cells)	4096	8000	10000	12000
Throughput (Mbit/s)	128.39	128.38	128.39	128.38
Fairness (1 기준)	0.58	0.58	0.58	0.58

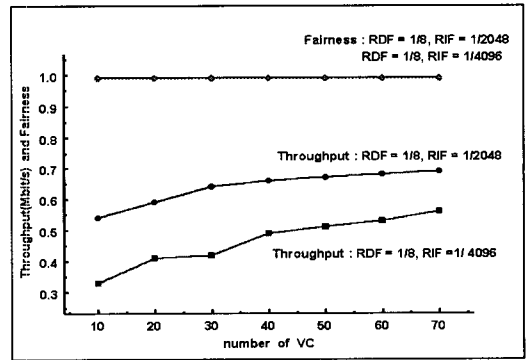
그러나, <표 3>에 나타난 결과와 같이 전체 버퍼 크기에서 EPD 임계치 변화에 따라 공정성과 처리율이 변화하는 것을 확인 할 수 있는데, 전체 버퍼 크기에 따라 최적화된 위치에 임계치를 결정하는 것이 처리율과 공정성 개선에 중요하다는 것을 알 수 있다.

<표 3> 버퍼 크기와 임계치 변화에 따른 처리율과 공정성

Buffer size	percent performance	percent						
		40%	50%	60%	70%	80%	90%	100%
4096	Throughput	96	107	118	124	119	113	107
	Fairness	0.31	0.43	0.52	0.55	0.57	0.66	0.69
8000	Throughput	125	126	128	128	129	129	122
	Fairness	0.60	0.60	0.63	0.61	0.65	0.64	0.54
10000	Throughput	126	126	128	128	125	124	109
	Fairness	0.60	0.67	0.61	0.67	0.60	0.61	0.72

5.2 ABR 상에서의 개선 방안

ABR 스위치 방식에서의 성능 개선 방안으로 파라미터 변화에 따른 개선을 나타낸다. ABR의 EFCI 스위치 방식에서 TCP 성능 개선을 위해서 두 파라미터 RIF와 RDF를 변경함으로써 안정된 처리율과 공정성 결과물 (그림 10)과 같이 보여주고 있다. 이러한 이유는 ACR를 천천히 증가시키다가 폭주 신호를 받을 때 급격하게 전송량을 감소시킴으로써 손실을 줄이고 서서히 증가하기 때문에 다른 TCP컨넥션이 전송할 수 있는 기회가 주어지기 때문이다. 그러므로 재전송량을 줄일 수 있고 공정성에 대해서도 안정적인 결과를 보인다. 그러나 실질적인 처리율 면에서는 전체 대역 용량을 1로 보았을 때 충분히 사용하지 못하고 있다. 여기서 처리율과 공정성 기준을 1로 설정하였으며, 처리율에서의 1은 전체 대역 용량을 나타낸다.



(그림 10) EFCI의 RDF와 RIF 변화에 따른 처리율과 공정성

ABR상의 ERICA를 사용한 ER스위치 방식에서 TCP 데이터는 손실이 하나도 없고 매우 안정적이며 공정성에 대해서도 우수한 성능을 보여주었다. 이러한 처리율과 공정성은 트랜스포트 계층의 TCP 혼잡 제어 방식 아닌 ATM 계층의 ERICA 알고리즘에 의해 제어를 받고 있음을 알 수 있다. 특히 송신원의 TCP의 윈도우 크기가 매 RTT 마다 2배씩 증가하면서도 손실이 없기 때문에 최대 윈도우의 크기 그대로 고정적으로 전송이 계속된다. 즉, 다시 말해서 일정한 전송 형태를 가지며, 고정된 윈도우 크기 때문에 스위치 버퍼 관리도 쉬어진다. ERICA상에서의 TCP 데이터는 불필요한 재전송과 셀 손실 없이 적당한 버퍼 관리 방식이

가능한데, 이 방식은 최악의 버퍼 상태 크기를 예측할 수 있다는 의미로, 크기는 $((3 \times c) \times RTT) \times BW$ 와 같이 접근된다[7]. 여기서 c 는 $O(\log N)$ 이며, N 은 VC의 수를 의미한다.

(그림 7)과 (그림 8)과 같이 전반적으로 좋은 성능을 보이고 있으나 스위치 파라미터인 AI의 값의 변화로 보다 나은 성능 개선을 가져올 수 있었다. 그 결과를 <표 4>에 나타내었다. 그러나 AI의 값은 n개의 셀이 처리되고 방출되는 평균 시간 간격이므로 너무 짧게 설정하면 활성화된 VC의 수를 잘못 추정하는 경우도 발생할 수 있을 것이다.

<표 4> AI(Averaging Interval)에 따른 처리율

스위치 파라미터 AI(Averaging Interval, time slot)	처 리 율(Mbit/s)
200	120.06
100	120.10
50	121.54
30	123.62

6. 결 론

본 논문에서는 시뮬레이션의 결과를 통해, 트랜스포트층의 TCP의 제어 방식과 ATM층의 ABR 및 UBR의 제어 방식의 차이점으로, 처리율과 공정성을 구하여 다음과 같은 결론을 얻었다.

- 1) UBR은 버퍼 크기에 성능이 좌우가 되고, UBR+EPD는 버퍼의 크기 변화 보다는 버퍼크기의 최적 임계치 결정이 중요하다는 것을 알 수 있었다.
- 2) ABR의 EFCI에서 VC 수의 증가에 따라 처리율과 공정성은 낮아졌으나, 최적화된 파라미터의 변화로 여러개의 VC가 전송률을 높일 기회를 갖게 되어 처리율의 증가와 높은 공정성을 확인할 수 있었다.
- 3) ABR의 ERICA 스위치 알고리즘에서의 TCP성능은 처리율이나 공정성의 면에서 높은 값을 가지며 매우 안정적인 결과를 나타내지만, 스위치 파라미터 AI 값에 따라 처리율이 변화됨을 알 수 있었다.

결론적으로 ABR 서비스에서는 EFCI 및 ER의 효율적인 파라미터 제어 운용과 UBR 서비스에서는 버퍼관

리를 통해 TCP/IP 성능을 향상시켜 개선된 처리율과 공정성을 얻을 수 있음을 알 수 있었다.

참 고 문 헌

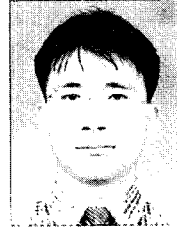
- [1] ATM Forum, ATM Traffic Management Specification Version 4.0, April 1996.
- [2] V. Jacobson, "Congestion Avoidance and Control," Proceedings of the SIGCOMM '88 Symposium, pp.314-332, August 1988.
- [3] YATS simulator for ATM networks, Dresden University of Technology, 1997.
- [4] L. Jaussi, M. Lorang, J. Nelissen, "A Detailed Experimental Performance Evaluation on TCP over UBR," IEEE ICATM '98, pp.214-223, June 1998.
- [5] S. Fahmy, R. Viswanathan, "ERICA Switch Algorithm : A Complete Description," AF-TM 96-1172, August 1996.
- [6] A. Romanow, S. Floyd, "Dynamics of TCP Traffic over ATM Networks," IEEE Journal on Selected Areas in Communications. Vol.13, No.4, pp.633-641, May 1995.
- [7] S. Kalyanaraman, Raj Jain, Sonia Fahmy, Rohit Goyal, "Performance and Buffering Requirements of Internet Protocols over ATM ABR and UBR Services," IEEE Communications Magazine, pp.152-157, June 1998.
- [8] D. Kouvatsos, 'ATM Networks : Performance and modelling and analysis volume 3,' 1st ED., Chapman & Hall, 1997.
- [9] O. Elloumi, H. Afifi, "RED Algorithm in ATM Networks," IEEE ATM '97 Workshop, pp.312-319, July 1997.
- [10] R. Goyal, et al., "UBR+ : Improving Performance of TCP over ATM-UBR Service," ICC '97, pp.1042- 1048, June 1997.
- [11] H. Li, et al., "A Simulation Study of TCP Performance in ATM Networks with ABR and UBR Services," Proc. IEEE INFOCOM '96, Vol.3, pp.1269-1276, May 1996.
- [12] H. Tzeng, K. Suil, "Performance of TCP over UBR in ATM with EPD and virtual queuing techniques," Computer Communications, Vol.21, No.12, pp.1070-107, Aug. 1998.



박 승 섭

e-mail : parkss@dolphin.pknu.ac.kr
1982년 경북대학교 공과대학
전자계산전공(공학사)
1984년 일본 일본대학 이공학연구
과(공학석사)
1993년 일본 동북대학(Tohoku Univ.)
(공학박사)

1984년~1986년 한국통신 연구원
1989년~1990년 일본동북대학 객원 교수
1998년 Philippine Ateneo de davao university, visiting prof.
1986년~현재 국립 부경대학교 컴퓨터멀티미디어공학부 교수
1999년~현재 부경대학교 컴퓨터멀티미디어 공학부장
관심분야 : 초고속통신망, 멀티미디어통신, ATM교환시
스템, 무선 ATM망



육 동 철

e-mail : net607@unicorn.pknu.ac.kr
1998년 동서대학교 컴퓨터공학과
졸업(공학사)
1999년~부경대학교 전자계산학
과 석사과정 재학중
관심분야 : 초고속 통신망, 인트라
넷, 무선 ATM