

단순 FIFO 기반 GFR 구현 기법에서의 TCP 성능 평가

박인용[○]

금오공과대학교 컴퓨터공학부
iypark@se.kumoh.ac.kr

TCP Performance over A Simple FIFO-based GFR Implementation Mechanism

Inyong Park

School of Computer Engineering, Kumoh National Institute of Technology

요 약

GFR 서비스는 ATM 네트워크에서 인터넷 트래픽을 효과적으로 처리하기 위하여 ATM 포럼에서 정의되었다. 단순 FIFO 기반 GFR 구현 기법은 가상 연결 단위의 계수기나 분리된 버퍼 관리가 필요치 않아 실용적이지만, QoS가 만족스럽지 않다. 본 논문은 이 구현 기법의 QoS를 개선할 수 있는 방안을 찾기 위해, 이 구현 기법의 문제점을 명확히 하고자 컴퓨터 시뮬레이션을 수행하고 그 결과를 분석하였다. TCP 트래픽에 따른 성능 평가를 위해 Tahoe와 New-Reno 두 버전을 각각 적용하였다.

1. 서 론

인터넷 트래픽을 ATM 네트워크에서 효과적으로 처리하기 위하여, GFR 서비스가 ATM 포럼에서 정의되었다[1]. GFR 서비스는 가상 연결이 전송하는 프레임의 크기와 burst의 크기가 연결 설정 단계에서 협의된 maximum frame size (MFS)와 maximum burst size (MBS)를 초과하지 않는다는 가정 아래서 예약된 대역폭 MCR을 가상 연결에게 보장하는 것을 목적으로 한다. 또한, GFR 서비스는 여분의 대역폭이 있을 때, 가상 연결들이 이 대역폭을 공평하게 사용하도록 허용한다. ATM은 가변 크기의 인터넷 패킷을 고정된 크기의 ATM 셀로 나누어 전송한다. ATM 스위치가 패킷의 일부 셀을 폐기하는 경우, 나머지 셀들만으로는 패킷을 복구할 없으므로, 그 셀들은 네트워크의 대역폭을 낭비하게 된다. 이러한 단편화에 의한 대역폭 손실을 막기 위해, GFR 서비스는 ATM 계층에서 패킷의 경계를 구분할 수 있는 AAL5 PDU를 인터넷 패킷을 실어 나르는 프레임으로 사용한다.

일찍이 ATM 네트워크에서 인터넷 트래픽을 위해 UBR 서비스와 ABR 서비스가 정의되었지만, UBR은 QoS 보장이 어렵고, ABR은 구현이 어려웠다. 그래서, UBR 서비스와 같은 간단한 스위치 기능만으로 ABR 서비스와 같은 QoS 보장을 제공할 수 있는, GFR을 위한 다양한 ATM 스위치 구현 방법들이 제안되었다[2,3]. 가장 간단한 구조를 갖는 기본 스위치 구현 방법은 프레임에 대역폭 보장에 적합한 것과 부적합한 것으로 나누는 F-GCRA 프레임 분류기와 가상 연결의 구분 없이 프레임 단위로 셀을 저장하는 EPD 방식의 버퍼 허용 알고리즘을 FIFO 버퍼에 적용하는 것이다. 이 방법은 구현이 쉽지만, 상대적으로 큰 대역폭을 예약한 가상 연결에게 예약한 크기만큼을 보장하지 못하는 것으로 알려져 있다[4]. 그래서, 가상 연결마다 버퍼 사용 계수기를 두고, 예약한 대역폭에 비례해서 버퍼를 할당하는 다양한 버퍼 허용 알고리즘들이 FIFO 버퍼를 갖

는 스위치를 위해 제안 되었다[3]. 또한, 가상 연결마다 별도의 버퍼를 두고 weighted fair queuing (WFQ)과 같은 전송률 보장 스케줄링을 적용하는 방법들도 제안되었다[2]. 그러나, 이 방법들은 GFR 서비스에 대한 만족스런 QoS를 제공할 수 있지만, 가상 연결마다 별도의 계수기와 분리된 버퍼가 필요하기 때문에, 구현 비용이 높고 운용 또한 쉽지 않다.

본 논문은 가상 연결 단위의 계수기나 분리된 버퍼 관리가 필요 없는 단순 FIFO 기반 GFR 스위치 구현 방법을 가지고 GFR 서비스의 QoS를 개선할 수 있는 방안을 찾기 위해, 컴퓨터 시뮬레이션을 통해 단순 FIFO 기반 GFR 스위치 구현 방법의 문제점을 명확히 하고자 한다. TCP 트래픽에 대한 성능 평가를 위해, TCP 혼잡 제어 알고리즘에 따른 두 가지 TCP 버전에 대해 시뮬레이션을 수행하였다. Tahoe 버전은 slow start와 fast retransmit 알고리즘을 채택하고 있으면, New-Reno 버전은 이 두 알고리즘 외에 개선된 fast recovery 알고리즘을 채용하고 있다.

2. 단순 GFR 스위치 구현 방법

FIFO 기반 단순 GFR 스위치 구현 방법은 그림 1에서처럼 F-GCRA 프레임 분류기와 단일 FIFO 버퍼에 두 개의 광역 임계값(global threshold)을 갖는 버퍼 허용 알고리즘으로 이루어진다.

2.1 F-GCRA 프레임 분류기

F-GCRA(I,T) 프레임 분류기는 가산 매개 변수 $I = 1/MCR$ 와 허용 매개 변수 $T \geq (MBS - 1) \times (1/MCR - 1/PCR) + CDVT$ 를 갖는 leaky bucket을 활용하여, 보장된 대역폭을 사용하기에 유효한 ATM 셀을 프레임 단위로 판정한다. 그리고, 보장된 대역폭 사용에 부적합하다고 판정된 프레임에 속하는 셀들의 cell loss priority (CLP) 비트 값을 1로 설정하는 tag를 수행한다. 버퍼 허용 알고리즘은 셀에 대한 버퍼 허용 여부를 판단할 때, CLP 비트 값에 따라 다른 기준을 적용할 수 있다.

leaky bucket은 가상 연결마다 별도의 상태 변수를 유지한다. 상태 변수로는 bucket의 채워진 정도를 가리키는 bucket 계수기 X, 보장된 대역폭 사용에 적합한 프레임에 속하는 가장 최근에 처리한 유효 셀의 도착 시간을 가리키는 LPT, 그리고, 현재 처리하고 있는 프레임의 적합성 여부를 가리키는 tag 등이 있다. F-GCRA 프레임 분류기의 자세한 동작은 다음과 같다.

초기화 $X = 0, LPT = Ta(1)$

가상연결 VC의 k 번째 셀 $Ta(k)$ 시간에 도착

$$X' = X - (Ta(k) - LPT)$$

if (the first cell of a frame)

if ($X' > T$) tag = true
else tag = false

endif

if (Not tag)

$I = 1/MCR$
 $X = \max(0, X') + I$
 $LPT = Ta(k)$

else

tag the cell

2.2 EPD 기법의 버퍼 허용 알고리즘

단순 GFR 스위치는 같은 출력 단자로 향하는 여러 가상 연결들이 함께 공유하는 단일 버퍼를 사용하여, 저장된 셀을 FIFO 스케줄링에 따라 출력 단자로 보낸다. 이러한 단일 FIFO 버퍼에서는 셀을 버퍼에 저장할지 혹은 폐기할지를 결정하는 버퍼 허용 알고리즘이 각 가상 연결의 링크 사용량을 조절한다. 단순 GFR 스위치는 같은 버퍼를 공유하는 모든 가상 연결에게 공통된 광역 임계 값 (global threshold)을 적용한다. 버퍼 사용량이 LBO threshold보다 작으면, 버퍼 허용 알고리즘은 도착하는 모든 셀들을 버퍼에 저장한다. 그러나, 버퍼 사용량이 LBO threshold를 넘어서면, 버퍼 허용 알고리즘은 F-GCRA 프레임 분류기에 의해 tag된 셀은 더 이상 버퍼에 저장하지 않고 폐기한다. 하지만, tag 안된 유효 셀은 여전히 버퍼에 저장한다. 버퍼 사용량이 HBO threshold를 넘어서면, 버퍼 허용 알고리즘은 tag 안된 유효 셀 역시 더 이상 버퍼에 저장하지 않고 폐기한다. 그러므로, HBO threshold는 버퍼의 크기인 MAX_Q에 근접하는 값으로 설정한다.

버퍼 허용 알고리즘은 각 threshold에서 셀의 저장 여부를 결정할 때 EPD (early packet discard) 기법을 사용한다. EPD 기법은 프레임의 경계를 고려하여 프레임 단위로 셀들의 버퍼 수용 여부를 결정한다. EPD 기법은 버퍼 사용량이 threshold를 넘으면, 새롭게 도착하는 프레임에 속하는 셀을 폐기하기 시작한다. 프레임을 폐기할 때는 그 프레임을 구성하는 첫 번째 셀부터 마지막 셀까지 모든 셀들을 폐기한다. 그러나, 버퍼 사용이 threshold를 넘기 전에 프레임의 첫 번째 셀을 이미 버퍼에 저장하였다면, 그 프레임에 속하는 나머지 셀들도 역시 버퍼에 저장한다.

3. 실험 결과 및 성능 평가

단순 GFR 스위치 구현 방법을 적용했을 때, ATM 네트워크에서

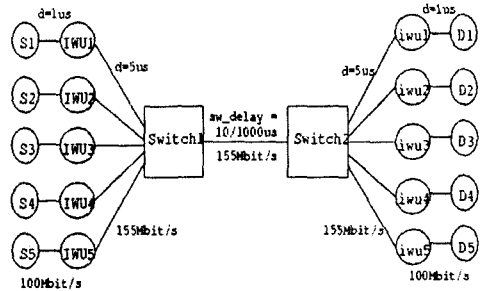


그림 2. 모의실험 환경

TCP 트래픽에 대한 GFR 서비스의 QoS를 평가하기 위해 NIST ATM 시뮬레이터를 기반으로 시뮬레이션을 수행하였다[4]. TCP 트래픽은 Tahoe 버전과 New-Reno 버전 대하여 각각 생성하였다.

3.1 실험 환경

시뮬레이션 환경은 그림 2에서 볼 수 있듯이, 5개의 송신 TCP와 5개의 수신 TCP가 100Mbit/s fast Ethernet을 통해 각각의 interworking unit (IWU)와 연결되어 있고, 각 IWU와 두 개의 ATM 스위치는 155.5Mbit/s의 OC-3c 링크로 연결되어 ATM 네트워크를 구성한다. 송신 TCP와 수신 TCP는 ATM 네트워크의 가상 연결을 통해 일대 일로 연결된다. 송신 응용 프로그램은 각자의 상대방에게 보낼 데이터를 무한히 가지고 있는 FTP 서비스로 가정한다. MSS는 1024 바이트로 고정하고, 최대 윈도우 크기는 64K 바이트로 한다. 1024 바이트 MSS를 갖는 TCP 세그먼트에 대한 ATM 네트워크에서의 MFS는 오버헤드를 포함해서, 23 ATM 셀이다. $((1024+20(TCP)+20(IP))+8(LLC)+8(AAL5))/48=22.5)$. 최대 버퍼 크기 MAX_Q는 3000 ATM 셀을 가지며, 버퍼 허용 알고리즘에서 EPD 기법을 위한 HBO threshold는 1500 ATM 셀로 한다.

3.2 성능 평가

WAN 환경을 고려하여, 두 ATM 스위치 사이의 전파 지연(propagation delay)을 1ms로 설정하여 시뮬레이션을 수행하였다. TCP 계층에서 중복되지 않은 완전한 프레임을 위해 전달된 ATM 셀의 양을 각 TCP 연결에 대한 유효 처리율(goodput)로 하였다. 또한, ATM 스위치에서 각 GFR 가상 연결에 대한 ATM 처리율을 측정하였다. 이 ATM 처리율은 F-GCRA에 의해 tag된 프레임에 의한 여분의 대역폭 사용량과 untagged 프레임에 의한 보장된 대역폭 사용량으로 이루어진다. 모든 GFR 가상 연결의 ATM 처리량의 총합은 링크의 사용율을 나타낸다.

실험에 사용된 다섯 개의 GFR 가상 연결은 각각 5, 10, 15, 20, 25 Mbit/s MCR을 예약 하여, 링크에 대해 총 75Mbit/s를 예약한다. 시뮬레이션 결과를 각 가상 연결이 기대하는 MCR과 여분의 대역폭 (RBW)을 가상 연결 수로 공평하게 나누었을 때 얻을 수 있는 추가분 FS (fair share)을 더한 MCR+FS와 비교하였다.

$$RBW = \text{Link_Capacity} - \sum MCRk$$

$$FS = 1/n \times RBW$$

각 가상 연결의 untagged 프레임이 사용한 대역폭이 MCR에 근접한다면, 예약한 MCR만큼의 최소 대역폭을 보장한다는 의미를 갖는다. 또한, ATM 처리율이 MCR+FS에 가까울수록 여분의 대역폭을 좀더 공평하게 사용한다는 것을 의미한다.

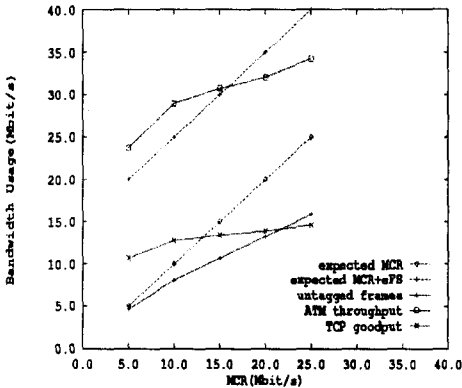


그림 3. Tahoe TCP 대역폭 사용량 (WAN)

그림 3에서 가상 연결이 예약한 MCR이 커질수록, untagged 프레임이 사용한 대역폭과 MCR의 격차가 더욱 커짐을 볼 수 있다. 이것은 단순 GFR 스위치 구현 방법이 MCR이 클수록 예약한 대역폭을 보장하는데 어려움을 갖는다는 것을 보여준다. 한편, TCP 처리율 역시 큰 MCR을 갖는 가상 연결이 MCR+FS에 못미치는 것을 볼 수 있다. 이 또한, 예약한 대역폭만큼을 보장하지 못하고 있음을 보여준다. 특이한 사항은 ATM 처리율에 비해 TCP 유효 처리율이 큰 격차를 보이는 것이다. 이것은 TCP Tahoe 버전의 혼잡 제어 알고리즘이 매우 비효율적으로 동작한다는 것을 보여준다. ATM 처리율에 비해 TCP 유효 처리율이 낮은 까닭은 ATM 셀에 의해 전송된 많은 프레임이 중복되었기 때문이다.

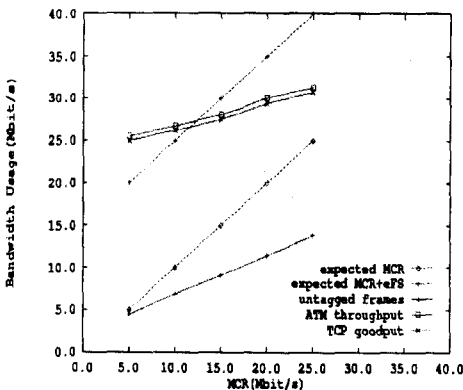


그림 4. New-Reno TCP 대역폭 사용량 (WAN)

그림 4를 통해 New-Reno 버전 역시 Tahoe 버전과 마찬가지로 큰 MCR을 갖는 가상 연결일수록 untagged 프레임에 의해 사용된 대역폭이 예약한 MCR에 미치지 못함을 볼 수 있다. 한편,

New-Reno에서의 TCP 유효 처리율은 ATM 처리율에 근접함을 볼 수 있다. 이것은 New-Reno의 혼잡 제어 알고리즘이 효율적으로 동작하여, 중복된 프레임의 전달을 많이 줄일 수 있었음을 보여준다. 또한, 작은 MCR을 갖는 가상 연결의 TCP 유효 처리율이 MCR+FS를 상회하는 반면, 큰 MCR을 갖는 가상 연결의 TCP 유효 처리율은 MCR+FS에 크게 못 미치는 것을 볼 수 있다. 이것은 작은 MCR을 갖는 가상 연결들이 큰 MCR을 갖는 가상 연결들보다 여분의 대역폭을 더 많이 사용하고 있음을 의미한다. 이것으로부터 단순 GFR 스위치 구현 방법은 작은 MCR을 예약한 가상 연결이 여분의 대역폭 사용에서 더 유리하도록 허용하고 있음을 알 수 있다.

4. 결론

본 논문은 가상 연결 단위의 계수기나 임계 값 (threshold)을 필요로 하지 않는 단순 GFR 스위치 구현 방법의 두 TCP 버전에 대한 성능 평가를 컴퓨터 모의실험을 통해 수행하였다. 두 TCP 버전 중 Tahoe 버전은 중복된 프레임의 전송으로 ATM 처리율과 TCP 유효 처리율에 큰 차이를 보였다. TCP 혼잡 제어를 위해 slow start와 congestion avoidance 만으로는 충분치 않음을 확인할 수 있었다. 반면 New-Reno 버전의 ATM 처리율과 TCP 유효 처리율은 거의 차이가 없는 것을 보아 중복된 프레임의 전송을 최소화하여, 효율적인 혼잡 제어를 수행하고 있음을 알 수 있었다.

한편, TCP 버전에 상관없이 더 큰 MCR을 갖는 가상 연결일수록 untagged 프레임이 사용한 대역폭이 예약한 MCR에 더 못 미치는 것을 볼 수 있었다. 이것은 단순 FIFO 기반 GFR 스위치 구현 방법이 갖고 있는 한계를 나타낸다. 즉, 상대적으로 가장 작은 MCR을 갖는 가상 연결을 제외한 다른 가상 연결에게 예약한 MCR 만큼의 대역폭을 보장하지 못하고 있는 것이다. 이러한 현상은 F-GCRA 프레임 분류기가 필요 이상의 많은 프레임들을 tag하고 있다는 것을 보여준다. TCP 트래픽 특성에 따라 적절한 양의 프레임만을 tag하도록 F-GCRA 프레임 분류기가 수정될 필요가 있겠다. 또한, tag된 프레임이 사용한 대역폭은 더 작은 MCR을 갖는 가상 연결이 더 큰 것을 확인할 수 있는데, 이 또한 EPD가 여분의 대역폭을 공평하게 할당하는데 한계가 있음을 보여준다. tag된 프레임들에 대한 공평성을 향상시키기 위한 방안이 앞으로 연구되어야 할 것이다.

참고문헌

- [1] The ATM Forum, "Traffic management specification 4.1", af-tm-00121.000, March 1999.
- [2] I. Andrikopoulos et al., "Providing rate guarantees for Internet application traffic across ATM networks", IEEE Communications Survey, 2(3), pp.2-13, 1999.
- [3] O. Bonaventure and J. Nelissen, "Guaranteed frame rate: a better service for TCP/IP in ATM networks", IEEE Networks, 15(1), pp.46-54, Jan/Feb. 2001.
- [4] Nada Golmie et al., "The NIST ATM/HFC network simulator", National Institute of Standards and Technology, 1998.