

ATM 망에서 인터넷 트래픽을 서비스하기 위한 효율적인 스케줄링 알고리즘에 관한 연구

(An Efficient Scheduling Algorithm for Internet Traffic over ATM Network)

金觀雄*, 裴晟桓**, 田炳實***

(Kwan-Woong Kim, Sung-Hwan Bae, and Byoung-Sil Chon)

요약

GFR 서비스 클래스는 ATM 네트워크에서 TCP/IP 트래픽을 효율적으로 서비스하기 위해 제안되었다. GFR은 VC에 프레임 단위의 최소 전송율을 보장하고 잉여 대역을 공평하게 할당하여야 한다. GFR 스위치 구현은 FIFO 큐잉 방식과 perVC-Queuing 방식으로 나눌 수 있다. 일반적으로 FIFO 큐잉 방식은 GFR이 요구하는 대역 보장을 하기 어렵기 때문에 perVC-Queuing이 적합하다. perVC-Queuing 구현 방식에서 스케줄링 알고리즘은 대역 보장과 공평성 성능 향상에 중요한 역할을 한다. 본 논문에서는 GFR 서비스를 위한 새로운 스케줄링 알고리즘을 제안한다. 제안된 알고리즘은 VC의 최소 전송율을 보장하고 공평성을 향상한다. 컴퓨터 시뮬레이션을 통해 제안한 알고리즘이 기존의 알고리즘 보다 TCP Goodput과 공평성에서 우수한 성능을 보였다.

Abstract

Guaranteed Frame Rate(GFR) service is intended to efficiently support TCP/IP traffic in ATM networks. The GFR service not only guarantees a minimum service rate at the frame level, but also supports a fair share of available bandwidth. The original GFR proposal outlined two switch implementation scheme : FIFO Queuing and perVC-Queuing. In general, it has been shown that FIFO Queuing is not sufficient to provide rate guarantees and perVC-Queuing with scheduling is needed. In perVC-Queuing implementation, scheduling algorithm plays key rule to provide rate guarantees and to improve fairness. We proposed a new scheduling algorithm for the GFR service. Proposed algorithm can provide minimum service rate guarantee and fair sharing to GFR VCs. Computer simulation results show that proposed scheduling scheme provide a much better performance in TCP Goodput and fairness than previous scheme.

Keywords : Scheduling, GFR, TCP/IP

* 正會員, 韓國標準科學研究院 人間情報그룹 Post. Doc. (Korea Research Institute of Standards and Science)

** 正會員, 漢麗大學校 멀티미디어情報通信工學科 (Dept. of Multimedia, Information and Telecommunication Eng., Hanlyo Univ.)

*** 正會員, 全北大學校 工科學 電子情報工學部 (Division of Electronic and Information Eng., Chonbuk Nat'l Univ.)

接受日字:2002年5月8日, 수정완료일:2002年8月12日

1. 서론

UBR(Unspecified Bit Rate)은 “best-effort” 서비스이기 때문에 다른 서비스가 사용하고 남은 잉여 자원이 부족할 경우에 원하는 서비스 품질을 얻기가 불가능하다. 이에 따른 문제를 해결하기 위해 프레임 기반의 데이터를 ATM 망을 통해 비교적 용이하게 전송하기 위해서 ATM 포럼에서는 새로운 ATM 계층 서비스로

GFR 서비스를 제시했다^[1].

GFR 서비스는 프레임 기반의 비 실시간 어플리케이션을 위해 제안되었으며, 응용계층으로부터 전달된 프레임 데이터에 대하여, 프레임이 최대 프레임 크기(MFS: Maximum Frame Size)와 최대 버스트 크기(MBS: Maximum Burst Size)를 넘지 않을 경우 MCR (Minimum Cell Rate)을 보장받아야 한다. 그리고 MCR 또는 MFS, MBS를 초과하여 전달된 프레임에 대해서는 MCR만을 보장하며 초과된 트래픽에 대해서는 가용 자원이 있을 경우, 각 사용자에게 공평히 나누어 전송하도록 정의하고 있다.

GFR의 데이터 트래픽은 AAL5 계층에 기초하고, 네트워크 스위치는 AAL5 프레임의 경계를 인식하도록 규정되어 폭주상황이 발생하면 단일 셀을 폐기하지 않고, 프레임에 속한 전체 셀을 폐기한다. 즉, 연결 설정 시 MFS, MBS, MCR을 협상하여, MCR에 대해서는 손실 없이 전송한다. 협상된 조건을 초과하여 전송된 프레임에 대해서 서비스 보장은 되지 않으며 UBR 서비스와 같이 "best-effort"로 전송하는 것이다^[1].

GFR 서비스가 UBR 서비스와 가장 다른 특징은 MCR을 보장하는데 있다. GFR 서비스를 지원하기 위해서는 크게 태깅(tagging), 버퍼관리(buffer management), 스케줄링 (scheduling)의 구성 요소가 필요하다. 태깅은 MCR을 보장하기 위해 프레임 단위로 순응검사를 수행하여 프레임이 MCR에 적합한 프레임과 부적합한 프레임으로 분류한다^[1]. 버퍼 관리는 네트워크 스위치의 버퍼에 프레임 저장을 제어하는 것으로 버퍼 관리는 버퍼 구현방식과 패킷폐기 방식으로 이루어지며, 버퍼 구현방식은 공유 버퍼방식^[2]과 각 연결당 논리적인 버퍼를 제공하는 perVC-Queuing 방식^[2]이 있으며 패킷 폐기 방식은 EPD [Early Packet Discard]^[3], PPD (Partial Packet Discard)^[3], RED(Random Early Packet Discard)^[4] 방식이 있다.

스케줄링은 연결된 VC들에 저장된 셀의 전송 순서를 결정하여 순서에 따라 셀을 전송하는 것으로 이를 담당하는 스케줄러는 버퍼에 저장된 데이터의 전송을 담당하며 GFR 서비스를 위해서는 모든 연결된 VC에 대해 MCR과 공평성을 제공해야 한다. GFR 서비스를 위한 스케줄링 알고리즘으로 WRR(Weighted Round Robin)^[5] 방식과 WFQ(Weight Fair Queuing)^[6] 방식이 있다. WFQ 방식은 각 연결당 요구하는 대역을 제공하고 잉여 대역을 공평히 할당할 수 있는 장점이 있는 반면, 구

현의 복잡도가 높고 스위치의 프로세싱 부하가 많이 요구되는 단점이 있다. WRR의 경우 각 연결에 가중치를 제공하여 가중치에 따라 셀을 서비스하는 방식으로 구현이 간단하고 적은 연산을 요구하므로 고속 스위칭에 적합한 알고리즘이다. 그러나 셀 지연이나 요구되는 대역을 정확히 제공할 수 없는 단점이 있다^[5,7].

본 논문에서는 기존의 GFR 서비스 방식에서는 셀 스케줄링 시, 셀의 CLP를 고려하지 않고 모든 프레임에 대하여 MCR 제공하므로^[5,6], CLP = 0인 프레임에 대하여 MCR를 제공할 수 없는 상황이 발생하는 단점을 보완하기 위해 CLP가 0인 프레임에 대하여 MCR을 보장하고, 높은 공평성을 제공하여 TCP의 Goodput을 향상시킬 수 있는 스케줄링 알고리즘을 제안한다.

II. GFR 서비스 구현 모델

GFR 서비스에서 각 연결 당 최소 전송률을 보장하고 대역의 공평한 할당을 위해서 그림 1과 같은 태깅, 버퍼 관리, 스케줄링의 세 가지 구성요소가 필요하다^[2].

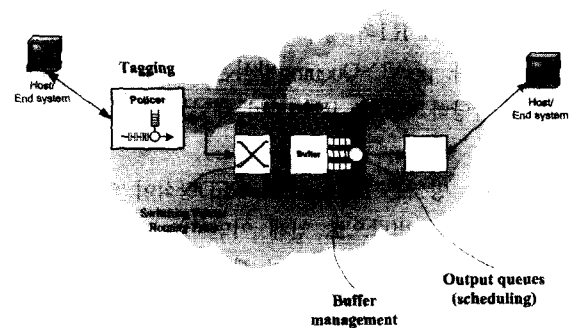


그림 1. ATM 네트워크의 태깅, 버퍼 관리, 스케줄러의 구조

Fig. 1. Tagging, buffer management and scheduler architecture in ATM network.

1. 태깅

ATM 셀이 망에 진입하기 전에 QoS(Quality of Service)에 적합한 셀과 부적합 셀을 구분하기 위해 부적합한 셀에 태깅을 하는 방법으로, 부적합 셀 헤더의 CLP 비트를 0에서 1로 변환한다. 태깅된 셀은 이미 계약한 트래픽을 위반한 것으로 간주한다. 따라서 버퍼 관리에서 CLP = 0인 셀이 CLP = 1인 셀보다 우선적으로 버퍼에 받아들인다. 태깅 방법은 망에 폭주가 발생하면 부적합한 프레임을 먼저 폐기함으로써, 폭주상황을 회피

하고, 가용 대역할당에 있어서 공평성을 향상시키는 데 사용될 수 있다. ATM 포럼 4.1에서는 F-GCRA (Frame-Based Generic Cell Rate Algorithm)^[1]을 통해 프레임에 대한 순응검사를 수행하여 적합 프레임과 부적합 프레임으로 구별할 수 있다.

2. 버퍼 관리

버퍼 관리는 네트워크 스위치의 버퍼에 프레임 저장을 제어하는 것으로 다양한 서비스의 셀 손실율을 보장하고 효율적으로 버퍼를 사용하기 위한 방법이다. 따라서 버퍼 관리는 스위치에 전송된 프레임을 저장할지 폐기할지를 결정하여 폭주상황에서 태깅된 셀은 우선적으로 폐기된다. 버퍼 관리는 버퍼 구현방식과 패킷폐기 방식으로 이루어지며, 버퍼 구현방식은 모든 연결이 버퍼를 공유하여 사용하는 공유 버퍼방식과 연결 당 가상의 큐를 제공하는 perVC-Queuing방식이 있으며, 공유 버퍼방식은 우선 순위가 높은 CLP = 0인 셀을 받아들이기 위해서는 버퍼에 빈 공간을 만들어야 한다.

빈 공간을 만들기 위해서는 버퍼내에 존재하는 낮은 우선 순위의 셀을 찾아서 폐기해야 하는데, 알고리즘의 복잡도와 공평성이 문제가 생기며 특히, 순응검사를 수행하지 않은 GFR1의 경우 각 연결의 MCR과 공평 할당이 어렵다. perVC-Queuing방식은 연결된 VC에 대해 버퍼를 할당하므로 MCR과 잉여 대역의 공평할당이 용이하다.

패킷폐기 방식은 PPD, EPD, RED방식이 있다. PPD 방식은 버퍼의 범람으로 인해 어떤 셀이 폐기되면 그 셀이 속한 프레임 내의 나머지 셀들을 식별하여 폐기함으로써 손실된 셀 이후의 불필요한 셀 전송을 막을 수 있도록 하는 방식이다. EPD방식은 불완전한 프레임이 전송되지 않도록 버퍼의 범람이 발생하기 전에 미리 전체 프레임을 버리는 방식이다. 현재의 버퍼의 길이가 미리 정해진 임계치를 넘을 경우 버퍼에 들어오는 프레임의 첫번째 셀이 폐기되고 동일한 프레임 내의 나머지 셀들도 계속해서 폐기된다. RED방식은 버퍼의 평균 길이가 임계치를 넘게 되면 버퍼의 평균 길이에 의해 정해진 폐기 확률에 따라 프레임을 폐기하는 방식이다.

3. 스케줄링

태깅과 버퍼 관리는 네트워크로 입력되는 프레임을 제어하는 반면, 스케줄링은 다음 홉으로 프레임을 전달하기 위한 큐기법으로 스케줄링은 버퍼 구현 방식과 밀접한 관계가 있다. 버퍼 구현이 공유 버퍼방식인 경

우, FIFO 큐에서는 프레임들은 셀이 입력되는 순서대로 전송되므로, 여러 연결들 사이에서 프레임들을 분리 할 수 없다. 반면에 perVC-Queuing 방식은 버퍼의 각 연결 당 큐를 논리적으로 분할함으로써 서비스 품질을 제공하기 쉽다.

perVC-Queuing 스케줄링 알고리즘은 WRR과 WFQ이 구현되었다. WRR 방식은 연결 당 MCR에 따른 가중치(Weight)를 설정하고, 스케줄러가 연결의 가중치의 크기에 따라 셀을 서비스하는 방식이다. 그림 2는 WRR 스케줄링 알고리즘이다.

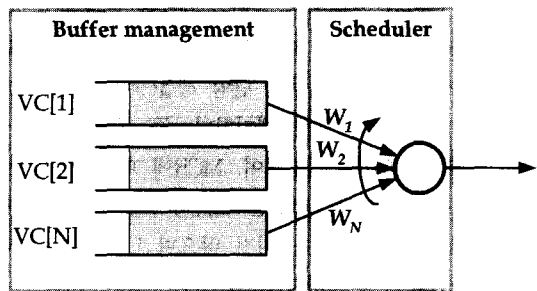


그림 2. WRR 스케줄러
Fig. 2. WRR Scheduler.

스위치의 버퍼 점유율이 LBO(Low Buffer Occupancy)이하 일 때는 각 VC의 가중치에 따라 대역폭을 할당하여 서비스하며 스위치의 버퍼 점유율이 LBO 이상이고 HBO (High Buffer Occupancy)이하이면 각 연결에 입력되는 태깅된 셀은 모두 폐기되므로, MCR을 보장할 수 있다. 그러나 WRR은 스케줄링 시 셀의 CLP를 고려하지 않으므로 CLP = 0인 프레임에 대한 MCR은 보장하기가 어려우며 공평성 면에서 개선할 부분이 있다^[7].

III. 제안된 스케줄링 알고리즘

GFR 서비스 클래스는 CLP가 0인 프레임에 대해 최소 전송율을 보장해야 한다. 초과된 CLP = 0인 프레임이나 CLP = 1인 프레임은 best-effort 방식으로 서비스하고, 폭주상황이 발생되면 스위치는 CLP = 1인 프레임을 우선적으로 폐기한다^[5,6]. 따라서 셀 서비스시 최소 전송율에 따른 CLP = 0인 셀과 CLP = 1인 셀을 구별하여 서비스할 필요가 있다.

GFR 서비스를 위한 기존 스케줄링 방식은 프레임 서

비즈니스 VC의 가중치에만 의존하여 동작하므로, CLP = 0인 프레임의 최소 전송률을 보장하지 못하는 단점이 있다^[5]. 따라서 제안한 알고리즘은 CLP가 0인 셀의 최소 전송률을 보장하기 위하여, 우선 순위를 사용한다. 우선 순위는 "High"와 "Low" 두 상태를 이용하여, 스케줄링 주기(scheduling cycle)동안 CLP = 0인 셀의 서비스율이 MCR이하인 VC는 높은 우선 순위인 "High"를 가지고, MCR 이상인 VC는 낮은 우선 순위인 "Low"를 가지게 하여 스케줄러가 셀을 서비스할 때 우선 순위가 "High"인 VC를 먼저 서비스함으로 CLP = 0인 프레임에 대하여 MCR을 보장 할 수 있다.

제안된 스케줄링 알고리즘 기법은 CLP = 0인 셀의 서비스율을 측정하기 위해, 먼저 각 VC에 입력되는 CLP = 0인 셀의 카운터 CI와 CLP = 0인 셀이 서비스될 때 사용되는 COUT 카운터 변수를 제공한다. ATM 스위치는 스케줄링 주기의 시작 시의 입력 카운터 CI를 0으로 초기화하고 CLP = 0인 셀이 도착함에 따라서 MCR의 가중치(W_{min}) 보다 작으면 CI를 증가하고, CLP = 0인 셀의 서비스 카운터 COUT도 따라서 증가하게 된다.

그림 3은 제안된 알고리즘에서 셀이 VC 큐에 도착할 때와 서비스 될 때 VC의 가중치와 CI, COUT의 갱신 과정을 보여준다.

VC의 MCR 가중치인 W_{min}은 스케줄링 주기 동안의 VC의 최소 전송율로 식 (1)과 같이 계산한다.

$$W_{min}[i] = \frac{MCR[i]}{C_{GFR}} \times T \quad (1)$$

여기서 MCR[i] : i-번째 VC의 MCR
 C_{GFR} : GFR 서비스를 위해 할당된 대역
 T : 스케줄링 주기 시간

VC의 가중치 W는 스케줄링 기간동안 VC가 전송할 수 있는 셀 수로 식 (2)와 같이 계산한다.

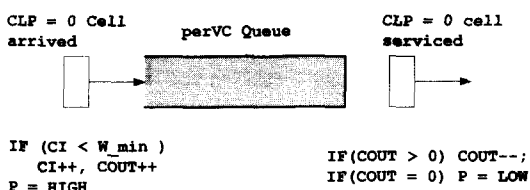


그림 3. 제안된 스케줄러의 동작
 Fig. 3. Behavior of proposed scheduler.

$$W[i] = \left(MCR[i] + \frac{\sum_{j=0}^N MCR[j]}{N} \right) \times \frac{T}{C_{GFR}} \quad (2)$$

여기서 N : 현재 서비스 받고 있는 GFR VC의 개수
 MCR[i] : i-번째 VC의 MCR
 C_{GFR} : GFR 서비스를 위해 할당된 대역
 T : 스케줄링 주기 시간

만약 VC의 COUT이 0보다 크다면, VC 큐에 CLP = 0인 셀이 존재함을 의미하므로 VC의 우선 순위는 "High"가 되고 스케줄러는 우선 순위가 "High"인 VC를 먼저 서비스한다. 우선 순위가 "High"인 VC의 CLP = 0인 셀이 서비스되면, COUT이 감소하게 되고 감소한 COUT이 0이면 해당 VC의 우선 순위는 "Low"로 바뀐다. 우선 순위가 "Low"로 낮은 VC들은 우선 순위가 높은 VC가 서비스된 후에 서비스를 받는다. 우선 순위가 "Low"인 VC에 새로운 CLP = 0인 셀이 도착할 때 우선 순위가 "High"로 높은 우선 순위를 가지게 된다. 그림 4는 제안된 알고리즘의 의사코드(pseudo code)이다.

```

//declaration
W : Weight of a VC
W_min : MCR weight of a VC
CI : counter of cell_has_CLP0 arrived in a scheduling period.
COUT : counter of cell_has_CLP0 when cell depart.
P : priority of a VC:
    if COUT > 0, P is High, else P is Low.

// Initialize phase or start new scheduling period
for (i = 0; i < N; i++)
{
    VC[i].W is "MCR + equalshare" or "MAX-MIN share";
    VC[i].CI = 0;
    VC[i].P = Low;
}

// When cell is arrived in VC(i)
if (cell_has_CLP0){
    if(VC[i].CI < VC[i].W_min){
        VC[i].CI++;
        VC[i].COUT++;
        VC[i].P = High;
    }
}

// When cell serviced
// service for High Priority VC
if (VC[i].P = High){
    VC[i] is served;
    VC[i].W--; // decrease Weight by 1
    if (cell_has_CLP0) {
        VC[i].COUT--; // decrease COUT by 1
    }
    if(VC[i].COUT = 0){ // priority change
        VC[i].P = Low;
    }
}

// when none of VC has High Priority
if (VC[i].W > 0){
    VC[i] is served
    VC[i].W--;
}
    
```

그림 4. 제안된 스케줄링 알고리즘 의사코드
 Fig. 4. Proposed scheduling algorithm pseudo code.

제안된 스케줄링 알고리즘은 VC의 CLP = 0인 셀의 서비스율에 따라 우선 순위를 두어 CLP = 0인 셀의 서비스율이 MCR 이하인 VC 큐의 셀을 우선 서비스하여 CLP = 0인 셀의 서비스율이 MCR을 충족시킬 수 있으며 공정성을 향상시킬 수 있어 기존의 스케줄링 알고리즘 비해 네트워크 수율과 공정성에서 개선 가능하다.

IV. 모의실험 환경

제안한 알고리즘의 성능평가를 위해, 컴퓨터 모의실험을 수행하였다. 시뮬레이션 프로그램은 C++로 작성했으며 이벤트 기반 구조로 구현되었다. GFR 서비스를 지원하기 위해서는 크게 태깅, 버퍼 관리, 스케줄링 세 가지의 구성 요소가 이루어지며 모의실험에 사용된 태깅 방식은 F-GCRA 알고리즘^[1]을 이용하였고, 버퍼 관리에서는 버퍼 구현 방식과 프레임 폐기 방식으로 perVC-Queuing 방식, EPD를 이용하였고 스케줄링 방식은 제안된 알고리즘을 적용하였다.

프레임 기반인 TCP/IP의 트래픽을 GFR을 통하여 서비스하는 경우, GFR은 흐름제어 메커니즘을 지원하지 않고, 흐름제어는 TCP계층에서 이루어진다. 모의실험에 사용된 TCP/IP 트래픽의 성능 평가를 위해서는 TCP의 흐름제어 메커니즘이 구현되어야 한다. TCP의 흐름제어는 Slow-start와 Congestion avoidance의 두 방식을 사용하여 제어하는데 TCP 연결이 설정되면, 흐름제어에 사용되는 두 변수 Congestion 윈도우(CWnd)와 Slow-start 임계치(Wth)를 사용한다.

TCP 송신 측에서 최초 세그먼트 전송시 CWnd의 크기는 1-세그먼트부터 설정한다. 수신 측으로부터 ACK 세그먼트를 받을 때마다 윈도우 크기를 두 배로 증가시킨다. 중복된 ACK 신호를 받거나 재전송 타이머가 만료되면, 폭주상태가 발생한 것으로 간주하여 CWnd를 줄여서 전송률을 감소시킨다. 중복된 ACK 세그먼트를 받는 경우는 전송된 세그먼트의 일부가 손실된 것으로 심각한 폭주상태는 아니므로, Congestion avoidance를 수행한다. 재전송 타이머가 만료된 경우는 전송한 모든 세그먼트가 유실되거나 수신측에서 전송한 ACK 세그먼트가 유실된 경우로, 이는 폭주상태가 심각하다고 판단하고, Slow-start를 수행한다. CWnd가 Wth에 도달하면, Congestion avoidance를 수행한다^[8, 9]. 그림 5은 TCP의 폭주제어 알고리즘의 흐름도이다. 시뮬레이션에

구현된 TCP계층은 Slow-start와 Congestion avoidance가 포함되어있다.

모의실험에 사용된 네트워크 환경은 그림 6에 보인바와 같이 10개의 TCP 송신원과 대응하는 10개의 TCP 수신원이 있으며 각 TCP 단말은 ATM 스위치와 링크로 연결되어 있다. 링크는 OC-3(155.5 Mbps), 지연시간은 0.3 m초이고 스위치와 스위치간의 지연시간은 3 m초로 큐잉 지연이 없다면 RTT(Round Trip Time)은 7.2 m초가 된다.

TCP의 환경변수 표 1과 같다. 송신원은 무한 크기의 파일을 TCP 세그먼트로 윈도우 크기만큼 전송하고, 수신원은 ACK 신호만을 전송한다. 재전송 방식은 Go-Back-N을 사용하였고, 폭주제어는 Slow-start와 Congestion avoidance 알고리즘을 사용한다. 그림 7은 VC의 MCR에 따른 TCP의 CWnd 크기의 변화를 보여준다.

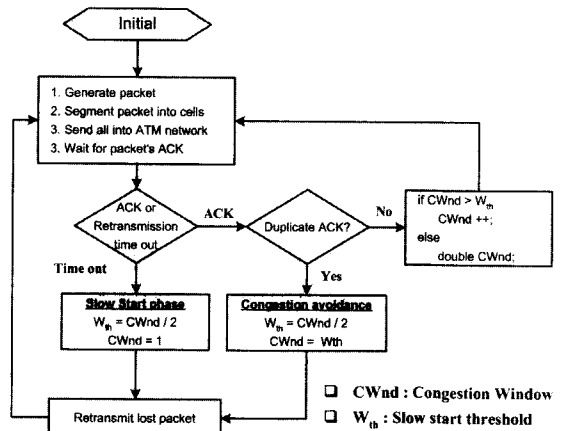


그림 5. TCP 폭주제어 알고리즘 흐름도
Fig. 5. Flow chart of TCP congestion control algorithm.

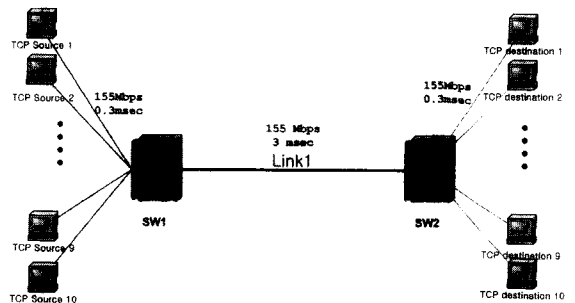


그림 6. 모의실험 네트워크 모델
Fig. 6. Simulation network model.

표 1. TCP 환경변수
Table 1. TCP parameters.

Maximum segment size	960 bytes
Max. Window size	115,200 bytes
Retransmission Timer	2*RTT
File size	Infinite

표 2. 스위치와 GFR 환경변수
Table 2. Simulation parameter of Switch & GFR.

Switch parameters	value
LBO	2,000 cell
HBO	6,000 cell
QMAX	8,000 cell
GFR source parameters	value
AAL5 PDU size	960 bytes
MFS	960 bytes
MBS	2*MFS

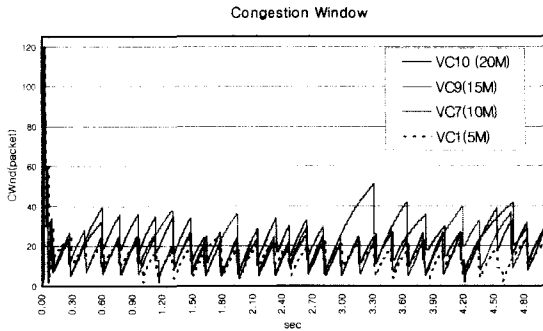


그림 7. MCR에 따른 TCP의 폭주 윈도우 크기
Fig. 7. Congestion window of TCP in different MCR.

TCP 계층에서 생성된 TCP 세그먼트는 AAL5 계층에서 PDU(Protocol Data Unit)로 변환되고, PDU는 ATM 계층으로 전달된다. ATM 계층에서는 수신된 패킷을 48 바이트로 쪼개어 5 바이트 헤더를 추가하여 ATM 셀로 변환한다. 표 2는 ATM 스위치와 GFR VC의 환경변수이다.

표 3은 각 VC당 PCR(Peak Cell Rate)과 MCR이다. 전체 MCR이 100 Mbps인 설정과 120 Mbps로 설정하여 최대 MCR에 따른 성능을 측정하였다. 각 VC가 셀을 전송하면 UNI(User-Network Interface)에서 F-

표 3. GFR VC의 MCR 및 PCR(Mbps)
Table 3. MCR & PCR of GFR VCs(Mbps).

		VC1	VC2	VC3	VC4	VC5	VC6	VC7	VC8	VC9	VC10
Config.1	PCR	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5
	MCR	5	5	5	5	10	10	10	15	15	20
Config.2	PCR	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5	155.5
	MCR	5	5	5	5	10	10	15	15	20	30

GCRA 테스트를 거쳐 순응검사를 수행한다. 즉 MCR 이하의 프레임은 CLP = 0으로, MCR 이상인 프레임은 CLP = 1로 변환되어 링크로 전송된다.

V. 모의실험 결과

GFR에서 TCP/IP 트래픽 서비스에 대한 중요한 성능 평가의 기준은 TCP의 Goodput과 공정성 지표이고 TCP의 Goodput은 TCP 계층에서 수신한 패킷을 전송률로 환산한 측정값을 공정성 지표는 각 VC들 사이에 잉여 대역을 공정하게 사용하는지 나타내는 척도로 잉여 대역을 나누는 방법에는 각 VC의 MCR에 비례해서 나누는 방법(Allocation proportional to MCR), 모든 VC에 똑같이 나누는 방법(MCR+equalshare)^[1, 10] 등이 있고, 공정성 정도를 알아보기 위해 식 (3)과 같이 공정성 지표를 계산한다.

$$\text{Fairness index} = \frac{\left(\sum_{i=1}^N x_i / f_i\right)^2}{N \times \sum_{i=1}^N (x_i / f_i)^2} \quad (3)$$

(여기서 x_i 는 VC_i 의 수율이며, f_i 는 VC_i 의 공정 할당량이다.)

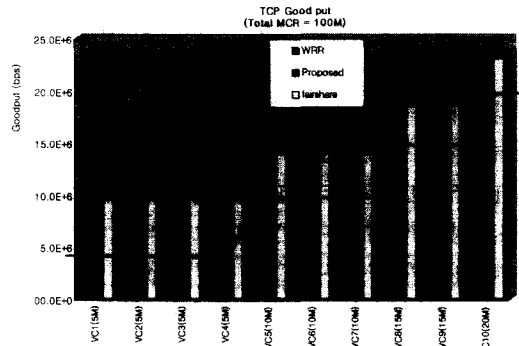


그림 8. TCP의 Goodput(Total MCR = 100 Mbps)
Fig. 8. TCP의 Goodput(Total MCR = 100 Mbps).

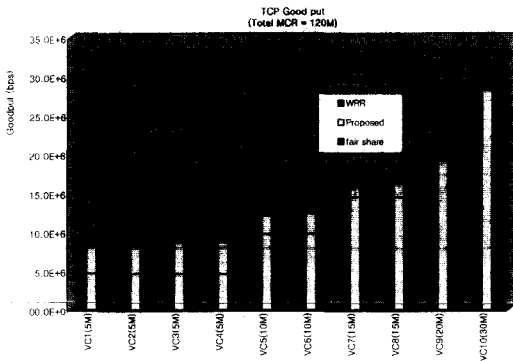


그림 9. TCP의 Goodput(Total MCR = 120 Mbps)
 Fig. 9. TCP의 Goodput(Total MCR = 120 Mbps).

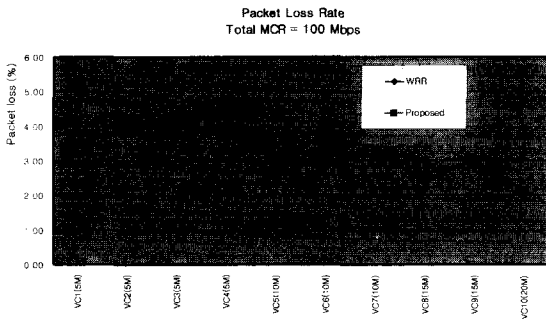


그림 10. 패킷 손실율
 Fig. 10. Packet loss rate.

그림 8은 전체 MCR의 총합이 100 Mbps일 때 제안한 스케줄링 방식과 기존의 WRR 방식을 적용했을 때 TCP의 Goodput이다. 공평율(fairshare)은 MCR + equalshare 방식으로 할당된 대역이고, 그래프를 가로지르는 선은 각 VC의 MCR 크기이다. 그림에서 MCR이 낮을 수록 Goodput이 좋으며, 높을수록 공평 할당율을 받지 못함을 나타낸다. 제안한 알고리즘의 경우 모든 VC들이 MCR 이상으로 서비스 받았으나 WRR 방식의 경우 MCR이 20 Mbps인 VC10은 MCR 미만으로 서비스 받았다.

그림 9는 전체 MCR이 120 Mbps일 때 TCP의 Goodput을 보여준다. WRR 방식의 경우에 MCR이 15 Mbps 이상인 연결들은 MCR 이하로 서비스 받는 반면, 제안한 방식의 경우는 MCR이 20 Mbps와 30 Mbps인 연결들만 MCR 이하로 서비스 받았다. 그림 10은 전체 MCR이 100 Mbps일 때, 각 VC별 패킷 손실율을 보여주는 것으로 제안한 알고리즘의 경우 모든 MCR에 대해 WRR 방식보다 낮은 패킷 손실율을 보여준다.

표 4. 스케줄링 알고리즘에 따른 성능

Table 4. Performance comparison of Proposed and WRR.

Scheduling scheme	Total MCR=100 Mbps		Total MCR=120 Mbps	
	WRR	Proposed	WRR	Proposed
Total Goodput	124.9Mbps	139.7Mbps	116.8Mbps	136.7Mbps
Fairness Index	0.9969404	0.9969572	0.9923253	0.9943569

표 4는 전체 Goodput과 공평성 지표를 보여주는 것으로 전체 Goodput은 모든 연결의 Goodput을 합한 것으로, ATM 스위치의 대역 사용율을 나타낸다. 제안한 방식이 WRR 방식보다 전체 Goodput에서 MCR의 총합이 100 Mbps일 때 약 15 Mbps의 성능 향상이 있고, 공평성에서도 앞섭을 보여준다. MCR의 총합이 120 Mbps일 때 WRR 방식은 MCR이 100 Mbps보다 8 Mbps정도 Goodput이 감소되나 제안한 방식의 경우 3 Mbps의 감소로 상대적으로 Goodput이 20 Mbps정도로 성능이 향상되었다. 컴퓨터 모의실험 결과 전체 MCR이 커질수록 공평성 지표와 전체 Goodput 성능이 떨어짐을 알 수 있다.

제안된 스케줄링 알고리즘은 기존 WRR보다 패킷 손실율을 줄여서 네트워크의 성능을 향상시켰으며 GFR이 요구하는 최소 전송율을 보장하였으며 기본 방식에 비해 잉여 대역의 공평한 할당을 통해서 공평성을 향상시켰다.

VI. 결론

본 논문에서는 ATM 계층 서비스로 새로이 제안된 GFR 서비스를 지원하기 위한 새로운 스케줄링 방식을 제안하고 그 성능을 컴퓨터 시뮬레이션을 통해 분석하였다. GFR 서비스 클래스는 CLP가 0인 프레임에 대해 MCR을 제공하고 여분의 대역을 연결된 VC에 공평하게 할당하여야 한다. 제안한 스케줄링 알고리즘은 버퍼에 저장된 셀이 전송될 때 CLP = 0인 셀에 대한 우선 순위를 두어 각 연결에 CLP = 0인 프레임에 MCR을 보장하고 공평성을 제공하였다. 컴퓨터 모의 실험을 통하여 기존 방식에 비해 공평성과 TCP의 Goodput이 향상됨을 확인하였으며 낮은 패킷 손실율로 패킷의 재전송을 줄여서 네트워크의 효율을 높였다.

참고 문헌

- [1] The ATM Forum Technical Committee, Draft TM 4.1 Traffic Management Specification, ATM Forum/BTD-TM-02.02, Dec. 1998.
- [2] I. Andrikopoulos, et al., "Providing Rate Guarantees for Internet Application Traffic across ATM networks," *IEEE Communications surveys*, vol. 2, no. 3, 1999.
- [3] S. Floyd and A. Romanov, "Dynamics of TCP Traffic over ATM Networks," *IEEE Journal on Selected Areas in Communications*, vol. 13, no. 4, pp. 633-641, 1995.
- [4] S. Floyd and V. Jacobson, "Random Early Detection Gateways for Congestion Avoidance," *IEEE/ACM Transactions on Networking*, 1(4), pp. 397-413, 1993.
- [5] J. Huang, et al., "A Simulation study of GFR Implementation," ATM Forum contribution, ATM 97-1035, Dec. 1997.
- [6] R. Guerin and J. Heinanen, "UBR+ Service Category Definition", ATM Forum contribution, ATM96-1598, Dec. 1996.
- [7] M. Katevenis et al., "Weighted Round Robin Cell Multiplexing in a General-Purpose ATM Switch Chip," *IEEE JSAC*. Vol. 9 No. 8 Oct. 1991.
- [8] Uyless Black, "TCP/IP & Related Protocols", McGraw-Hill. 1994.
- [9] S. Pappu and D. Basak, "TCP over GFR Implementation with Different Service Category : A Simulation Study," ATM Forum Contribution ATM97-0310, April 1997.
- [10] The ATM Forum Technical Committee, "ATM Forum Traffic Management Specification, Version 4.0", ATM Forum/AF-TM-0056.000, April 1996.

저자 소개

金 觀 雄(正會員) 第38卷 SD編 第4號 參照

裴 晟 桓(正會員) 第38卷 SD編 第4號 參照

田 炳 實(正會員) 第38卷 SD編 第4號 參照