

ATM 네트워크에서 TCP 성능향상을 위한 평균 전송을 기반의 버퍼관리 알고리즘

정회원 김관웅*, 이창기**

A Rate-Based Buffer Management Algorithm to Improve TCP Performance over ATM networks

Kwan-woong Kim*, Chang-Ki Lee** *Regular Members*

요약

본 논문에서는 perVC-Queuing 기반의 버퍼관리 알고리즘을 제안한다. 제안된 알고리즘은 서비스율 측정과 동적 VC 임계치를 사용하여 GFR VC에 네트워크 자원을 공정하게 분배하고 모든 VC가 요구하는 MCR을 보장한다. 컴퓨터 시뮬레이션결과 제안된 알고리즘이 GFR 서비스 클래스가 요구하는 QoS를 보장하고 TCP의 성능을 향상하였다.

Key Words : Buffer management; TCP/IP; IP over ATM, scheduling; congestion control

ABSTRACT

In this paper, we proposed a new buffer management algorithm using perVC-Queuing discipline. Proposed algorithm uses service rate estimation and assigns dynamic perVC threshold to each VCs. Service rate estimation and dynamic perVC threshold combined with global threshold provide nearly optimal throughput and improve fairness performance of network resource among GFR VCs as well as guarantee MCR of all VCs. From simulation results, we demonstrate the proposed scheme fulfills the requirement of GFR service as well as improves the TCP throughput.

1. 서론

초고속 네트워크 인프라가 구축되고, WWW, FTP, 전자상거래, 이메일 등의 서비스가 보편화되면서 인터넷 트래픽이 급증하게 되었다. 따라서 인터넷 트래픽과 같은 "best-effort" 트래픽을 ATM 망에서도 효율적으로 서비스를 제공하기 위해서 ATM Forum에서는 새로운 ATM 계층 서비스로 GFR(Guaranteed Frame Rate) 서비스를 제시했다^[1]. GFR은 인터넷 트래픽의 효율적인 서비스를 위해 최소 전송율을 보장하며 잉여대역을 각 연결된 VC

에 공평히 할당하여야 한다.

기존의 UBR(Unspecified Bit Rate)와 가장 큰 차이점은 UBR은 어떠한 서비스 품질도 제공하지 않은 반면 GFR 서비스는 호의 연결 설정시 최소 셀 전송율(Minimum cell rate: MCR)와 최대 프레임 사이즈(Maximum Frame Size: MFS), 최대 버스트 길이(Maximum Burst Size: MBS), 최대 셀 전송율(Peak Cell Rate: PCR)의 4가지 트래픽 파라미터를 협상하여, 프레임 레벨에서 VC에 최소 셀 전송율을 보장해야 한다.

GFR 서비스가 요구하는 QoS를 보장하기 위해서 Goyal은 태깅(Tagging), 버퍼관리(Buffer manage-

* 한국표준과학연구원 정보전산그룹(kkw@kriss.re.kr), ** 서남대학교 컴퓨터정보통신학과(lck@seonam.ac.kr)
 논문번호 : 030078-0225, 접수일자 : 2003년 2월 25일

ment), 스케줄링 (Scheduling)의 3가지 기법을 제시했다. 태깅은 MCR을 보장하기 위해 프레임 단위로 순응검사를 수행하여, 프레임이 MCR에 적합한 프레임과 부적합한 프레임으로 분류한다. 기존의 버퍼관리 방식은 주로 FIFO 버퍼를 이용하여 GFR 서비스를 지원하도록 하는데^[2], 버퍼에 최대 이용과 최소 이용에 대한 두 개의 임계치를 적용하는 EPD(Early Packet Discard)방식이나 PPD(Partial Packet Discard), RED(Random Early Detected) 등의 패킷폐기 방식을 통해 VC별로 프레임을 폐기함으로써 각 VC별 최소 전송률과 잉여 대역 사용에 대한 공평성을 보장하는 방식이다^[3, 4]. 대표적인 방식으로는 Double-EPD방식이나 DFBA(Differential Fair Buffer Allocation)방식이 있다^[5].

스케줄링은 연결된 VC들에 저장된 셀의 전송 순서를 결정하여 순서에 따라 셀을 전송한다. 이를 담당하는 스케줄러는 버퍼에 저장된 데이터의 전송을 담당하며, GFR 서비스를 위해서는 모든 연결된 VC에 대해 MCR과 공평성을 제공해야 한다. GFR을 구현하기 위한 스케줄링 기법은 FIFO Queuing 방식과 perVC-Queuing 방식이 있다.

FIFO Queuing 방식은 하나의 FIFO queue를 여러 개의 VC가 공유하는 방식으로 가장 빨리 도착한 패킷이 가장 빨리 서비스 받는 단순한 기법으로 VC들의 패킷 전송을 분리해서 서비스할 수 없으므로 GFR이 요구하는 최소 전송률 보장 및 공평성을 제공하기가 어렵다. 그러나 구현하기가 쉽고 단순한 방식이어서 FIFO 큐에서 성능을 향상시키기 위한 버퍼관리 기법이 많이 연구되고 있다. perVC-Queuing은 각 VC에 논리적인 큐를 할당하는 방식이다. 스케줄러가 스케줄링 알고리즘에 따라 서비스 받을 VC를 선택하여 패킷을 서비스한다. GFR 서비스를 위한 스케줄링 알고리즘으로 WRR (Weighted Round Robin)^[6] 방식과 WFQ(Weighted Fair Queuing)^[7] 방식이 있다. WRR은 각 VC에 MCR에 따른 가중치를 할당하여 가중치에 따라 서비스하는 방식으로 Pappu가 GFR 서비스에서 WRR을 적용하여 시뮬레이션을 통해서 성능을 평가하였다. 시뮬레이션 결과 WRR 알고리즘이 VC의 MCR을 보장할 수 있었다. WFQ 방식도 WRR과 같이 광범위하게 사용되는 스케줄링 알고리즘으로 Bonaventure가 WFQ 방식과 perVC-Queuing 기법을 적용하여 성능을 평가하여 TCP의 수율을 향상 시켰음을 증명하였다^[8]. 이와 같이 perVC-Queuing 방식은 GFR이 요구하는 QoS(Quality of Service)를

제공할 수 있지만, 스케줄러가 필요하여 추가비용이 요구된다.

GFR의 구현을 위해 ATM 스위치에서 GFR이 요구하는 최소 전송률 보장 및 높은 공평성을 제공하기 위한 버퍼 관리 방식에 대한 많은 연구가 이루어지고 있다. GFR에서 버퍼 관리는 네트워크 대역을 효율적으로 사용하고 폭주상태를 해결하는 기능 뿐 아니라, VC별 데이터 트래픽의 수용을 제어하여 VC에 할당된 MCR을 보장하고 공평성을 향상시키는 기능을 수행한다. GFR에서 TCP 성능을 향상시키기 위해서 Double-EPD^[3], DFBA^[5], WFBA (Weighted Fair Buffer Allocation)^[4]와 같은 FIFO 기반의 버퍼관리 알고리즘이 ATM 네트워크에서 GFR를 지원하기 위해서 제안되었다. 기존의 버퍼관리 알고리즘은 TCP 트래픽의 "burstness" 특성으로 인해 CLP가 0인 패킷의 손실 없는 전송만으로는 GFR이 요구하는 최소 전송률 보장과 공평성을 제공하기 어렵다. GFR의 대표적인 버퍼 관리 알고리즘인 Double-EPD나 DFBA의 경우 버퍼의 셀 점유율이 임계치 이상일 때 셀 폐기를 수행하므로 임계치 이하에서 일부 VC가 할당된 버퍼 공간 보다 많은 셀을 버퍼에 받아들일 경우 공평성이 나빠지는 단점이 있다. 또한 연결된 VC의 지연 시간이나 패킷의 크기가 서로 다른 경우에도 최소 전송률 보장과 공평한 대역 할당하지 못한다^[9].

본 논문에서는 perVC-Queuing 기반의 버퍼관리 알고리즘을 제안한다. 제안한 알고리즘은 동적 임계치와 VC의 서비스율을 측정하여, CLP = 1 패킷을 적절히 폐기함으로써 GFR이 요구하는 QoS를 제공한다. 특히 TCP 기반의 인터넷 트래픽을 서비스하는 경우, TCP연결의 성능은 패킷의 손실율과 패킷의 왕복시간에 큰 영향을 받는다. 제안한 알고리즘과 기존 알고리즘을 시뮬레이션을 통하여 비교 분석한 결과, 제안한 알고리즘의 경우 패킷을 적절하게 폐기하여 서로 다른 RTT와 TCP의 MSS가 다른 환경에서도 모든 VC의 MCR을 보장하고 높은 공평성을 제공하였다.

II. GFR 구현방식

2.1 태깅과 FIFO 큐를 사용한 GFR 구현방식

Guerin이 제안한 방식으로 Double-EPD라고 불리며, 두 개 전역 임계치와 패킷의 태깅정보만을 사용

해서 트래픽을 관리한다^[3]. 프레임이 UNI(User-Network Interface)의 F-GCRA를 통과하면 MCR이하의 패킷의 CLP는 0으로 설정되고, MCR을 초과한 패킷의 CLP는 1로 설정이 된다. Double-EPD 방식에서는 두 개의 임계치 LBO(Low Buffer Occupancy)와 HBO(High Buffer Occupancy)를 사용하는데 그림 1과 같이 FIFO 큐에 대기하고 있는 전체 셀 수(QT)가 LBO를 초과하면 우선순위가 낮은 CLP = 1인 패킷을 폐기하고, HBO를 초과하면 모든 패킷을 폐기한다.

이와 같이 태깅과 FIFO 큐를 이용하여 GFR을 구현하는 경우, 구현은 간단하지만 잉여 대역폭을 각 VC들에 공평히 분배할 수 없는 문제점이 있으며, 특히 TCP와 같은 버스티한 특성의 트래픽을 서비스할 경우 VC의 MCR을 보장하기 힘든 단점이 있다.

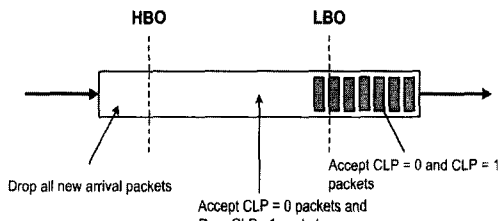


그림 1. 태깅과 FIFO 큐를 사용한 GFR 구현방식

2.2 perVC-Accounting을 사용한 GFR 구현방식

perVC-Accounting 방식은 버퍼를 공유하는 VC마다 MCR에 따른 버퍼 점유량을 할당하여 각 VC의 버퍼점유량이 자신의 할당량을 초과하지 못하도록 관리함으로써 각 VC가 자신의 fair share를 초과해서 대역을 사용하는 것을 막을 수 있기 때문에 공평성을 높이는 기법으로, DFBA나 WFBA 등과 같은 알고리즘이 연구되었다. perVC-Accounting 기법은 perVC-Queuing 기법보다 구현의 복잡도가 낮고 FIFO 큐와 태깅을 사용한 방식보다는 우수한 성능을 보여주나 GFR이 요구하는 모든 QoS를 만족하는 어렵다.

2.3 perVC-Queuing with threshold를 사용한 GFR 구현방식

GFR VC별로 버퍼 점유량을 두는 방식은 단지 셀을 폐기할 때 공평성만을 보장한다. ATM Forum TM 4.1에 근본적으로 각 VC에 대한 수용의 공평성을 보장하기 위해서 perVC-Queuing 방식과 스케줄러를 결합한 방식이 제안되었다^[5, 10].

perVC-Queuing 방식은 그림 2와 같이 VC별로 큐를 두어 각 VC별 큐에 셀을 저장하여 출력 부분에서 WFQ방식으로 셀을 전송한다. 각 VC별로 임계치 $T[i]$ 를 두어 잔여 임계치와 결합하여 패킷을 수용할지 여부를 판단한다. 그림 2는 perVC-Queuing with threshold 기법을 사용한 방식을 보여준다. 여기서 VC별 임계치 $T[i]$ 는 VC의 MBS의 배수로 설정한다. VC별 큐잉 방식은 버퍼에 셀을 저장할 때와 실제 전송할 때 모두 공평성을 보장할 수 있지만 WFQ 방식 스케줄러의 복잡도와 연결된 모든 VC마다 큐를 할당하므로 구현하는데 비용이 많이 드는 단점이 있다^[10].

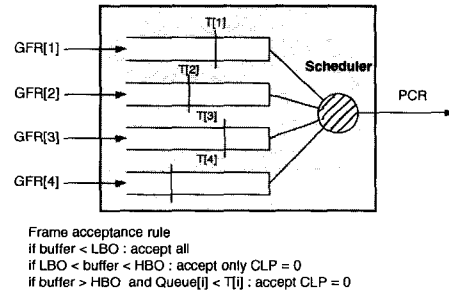


그림 2. perVC-Queuing with threshold

III. 제안된 버퍼관리 알고리즘

본 논문에서는 네트워크 효율을 높이며 높은 공평성을 제공하는 버퍼 관리 알고리즘을 제안한다. 제안된 버퍼 관리 알고리즘은 연결된 VC에 동적 임계치를 할당하고 VC의 서비스율을 측정한다. 동적 임계치는 전체 버퍼의 셀 점유량에 따른 VC의 할당된 버퍼공간이다. VC의 버퍼 점유량이 동적 임계치보다 클때 VC의 서비스율을 측정하여 할당된 대역보다 많은 서비스를 받은 VC의 패킷을 우선 폐기하여 할당된 대역보다 적은 서비스를 받은 VC의 버퍼 공간을 확보하여 공평하게 대역을 할당한다.

따라서 동적 임계치는 버퍼를 공유하는 VC의 버퍼 점유율을 조절함으로써 네트워크 효율을 향상시킬 수 있고 VC의 서비스율과 조합하여 패킷폐기

여부를 결정하여 TCP의 버스트 트래픽에 대하여 예약된 대역을 보장할 수 있으며 여분의 대역을 공정하게 할당하여 공평성을 향상한다.

3.1 VC 별 임계치

제안된 버퍼 관리 알고리즘은 Double-EPD 방식과 같이 두 가지 전역 임계치 LBO와 HBO 사용하며 버퍼의 공평한 할당을 위해서 VC에 동적 임계치를 할당한다. 동적 임계치는 전체 버퍼의 셀 점유율에 따라 VC에 할당된 버퍼 공간으로 할당된 대역보다 많은 대역폭을 사용하는 VC의 버퍼 공간을 제한하여 공평성을 향상시키는데 사용하며 식 (1)과 같이 계산한다. 스위치는 GFR VC에 현재 FIFO 버퍼 점유량과 VC에 할당된 대역에 따라 버퍼의 공간을 할당한다.

$$Th_i = \min(\max(QT, LBO), HBO) / N \quad (1)$$

여기서 QT : 버퍼에 대기하고 있는 셀 수(버퍼 점유량)
 N : VC의 갯수
 Th_i: i-번째 VC의 동적 임계치

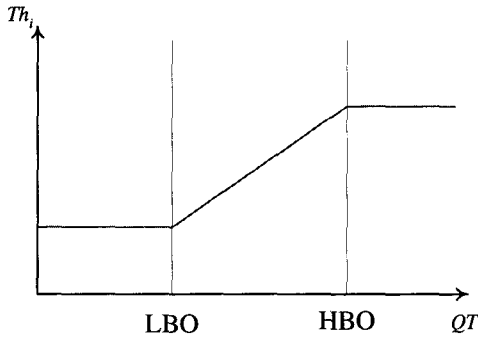


그림 3 버퍼 점유량에 따른 Th_i의 추이

3.2 VC의 서비스율 측정

VC 서비스 속도의 측정은 할당된 대역폭을 초과한 VC에 대해서 동적 VC 임계치와 결합하여 패킷 폐기 결정에 사용된다. ATM 스위치에서 i-번째 VC의 평균 서비스율은 식 (2)와 같이 지수 평균(exponential average)방법으로 측정하며 패킷의 첫 번째 셀이 도착할 때마다 측정이 이루어진다.

$$r_i = (1 - \alpha) \frac{l_i^k}{T_i^k} + \alpha \cdot r_i \quad (2)$$

여기서 r_i는 i-번째 VC의 서비스 속도이고, l_i^k는

k번째 패킷의 크기이다. T_i^k는 패킷의 도착 간격으로 T_i^k = t_i^k - t_i^{k-1}, α는 지수 가중치며 0에서 1사이의 값을 가진다.

w_i는 i-번째 GFR VC의 가중치로서 GFR VC에 할당된 대역과 같으며 식 (3)과 같이 계산한다.

$$w_i = MCR_i + \frac{C_{GFR} - \sum_j^{ActiveVC} MCR_j}{N} \quad (3)$$

여기서 C_{GFR} : 출력 버퍼에서 GFR에 할당된 대역
 MCR_i : i-번째 VC의 MCR

3.3 제안된 패킷 폐기 알고리즘

perVC-Queuing 기반의 제안된 버퍼 관리 알고리즘에서는 짧은 기간의 전송율인 GFR VC의 버퍼 점유율 Q_i 임계치 Th_i, 긴 기간의 전송율 파라미터인 가중치 w_i를 지수평균으로 측정한 평균 전송율 r_i를 이용하여 패킷의 폐기를 결정한다.

- CLP = 1인 패킷이 도착할 경우 다음과 같이 패킷 폐기 여부를 결정한다.
 - 경우 (1) : 버퍼의 셀 점유율이 LBO 미만일 때 GFR VC의 버퍼 점유율 Q_i가 임계치 Th_i보다 크고 r_i값이 w_i보다 크면, 할당된 대역폭 이상으로 서비스 받은 VC 이므로 패킷은 폐기된다.
 - 경우 (2) : 버퍼의 셀 점유율이 LBO 이상일 때 GFR VC의 버퍼 점유율 Q_i가 임계치 Th_i보다 크면, 패킷은 폐기된다.
 - 경우 (3) : 버퍼의 셀 점유율이 HBO 이상 일 때 패킷은 폐기된다.
- CLP = 0인 패킷이 도착할 경우 다음과 같이 패킷 폐기 여부를 결정한다.
 - 경우 (4) : 버퍼의 셀 점유율이 LBO 미만이면 패킷은 버퍼로 수용된다.
 - 경우 (5) : 버퍼의 셀 점유율이 LBO 이상이고 HBO 이하일 때, GFR VC의 버퍼 점유율 Q_i가 임계치 Th_i보다 크고 r_i값이 w_i보다 크면, 해당 VC는 QoS를 위반한 것으로 간주하고 패킷은 폐기된다.
 - 경우 (6) : 버퍼 점유율이 HBO 이상이면 심각한 폭주상태이므로 새로 도착하는 모든 패킷은 모두 폐기한다.

제안된 알고리즘의 의사코드는 다음과 같다. GFR

서비스 구현 연구에서 perVC-Queuing기반의 버퍼 관리 기법을 사용하는 경우 스케줄링 기법은 기존의 WRR 방식을 적용하였다.

Variables definition

VCQ[i]: i-th GFR VC Queue
 Q[i] : i-th GFR VC Queue occupancy
 Th[i] : i-th GFR VC Threshold
 r[i] : Service rate of i-th GFR VC
 PS[i] : Packet dropping flag
 w[i] : Weight of i-th GFR VC
 QT : Total buffer occupancy
 LBO : Low buffer occupancy threshold
 HBO : High buffer occupancy threshold
 QMAX : Buffer size
 T : Scheduling period

first cell of a frame arrived

```

if(cell_has_CLP1)
if((r[i]>w[i]) and (Q[i] > Th[i])) /*
drop CLP = 1 packet */
    discard cell;
    set PSi = 1

else if(QT > LBO and (Q[i] > Th[i])
/* buffer occupance is above LBO and
flow i's fairshare exceed.*/
    discard cell;
    set PS[i] = 1
    
```

```

else
    Evaluate_ServiceRate(r[i]);
/* accept cell to perVC Queue */
    VCQ[i].insertTail(cell);
    PS[i] = 0;
    QT++;
    Q[i]++;
    
```

subsequence cell of a frame arrived

```

if((PS[i] = 0) and (QT < QMAX))
/* accept packet */
    VCQ[i].insertTail(cell);
    Q[i]++;
    
```

```

    QT++;
else
    discard_cell;
    PS[i] = 1; /* makes sure
subsequence cells are discards */
    
```

IV. 시뮬레이션 및 성능 평가

제안된 버퍼 관리 알고리즘의 성능평가를 위해 GFR 시뮬레이션 프로그램을 개발하였다. 개발된 GFR 시뮬레이션 프로그램은 이벤트 기반의 시뮬레이션으로 이벤트를 주고 받아 수행되는 링크 큐, 스위치 등의 컴포넌트들로 구성이 된다. 시뮬레이션에 사용된 네트워크는 일반적으로 GFR 성능평가에 사용되는 모델로서 IP 라우터가 ATM 스위치와 직접 연결된 pear-to-pear 구성으로, 두 대의 ATM 스위치에 5대의 라우터가 결합된 네트워크로 그림 4 과 같다. 각 라우터는 5개의 TCP가 연결되고 각 TCP는 greedy 소스로 연결 기간동안 무한 크기의 파일을 전송한다.

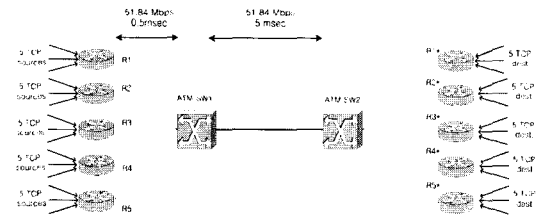


그림 4. 시뮬레이션 모델 I

TCP의 MSS는 기본 960 byte로 20개의 ATM 셀을 이루며 TCP의 시뮬레이션 파라미터는 표 1과 같다. 송신원과 수신원은 CLP = 0인 패킷만 전송한다. 전송된 패킷은 UNI의 F-GCRA 테스트를 통해 태깅이 수행된다^[11]. 네트워크의 모든 링크는 SONET OC-1인 51.84 Mbps이며 ATM 스위치간의 지연은 5 msec이고, 라우터와 ATM 스위치간의 지연은 0.5 msec이다. 그림 4의 왼쪽에 위치한 라우터들의 TCP는 송신원이고, 오른쪽에 대응하는 라우터의 TCP는 수신원이다. 시뮬레이션에 사용된 TCP는 New Reno버전으로, Slow-start, Congestion avoidance와 Fast-recovery, Fast-retransmit 알고리즘^[11]이 포함된다. ATM 스위치의 버퍼용량은 4K

셀, HBO는 3K 셀이고 LBO는 1K 셀로 설정하여 시뮬레이션을 수행하였다.

GFR에서 TCP/IP 트래픽 서비스에 대한 중요한 성능 평가의 기준은 VC 수율과 공평성 지표 (fairness index)이다. 공평성 지표는 각 VC들 사이에 잉여 대역을 공평하게 사용하는지를 나타내는 척도로 공평성 측정을 위해 식 (4)과 같이 공평성 지표를 계산한다^[12].

$$fairness\ index = \frac{\left\{ \sum_{i=1}^N x_i / f_i \right\}^2}{N \times \sum_{i=1}^N (x_i / f_i)^2} \quad (4)$$

여기서, x_i : VC_i의 수율, f_i : VC_i의 공평 할당량
N : VC의 개수

그림 5는 LBO를 1K 셀로 설정했을 때, MCR에 따른 TCP goodput의 시뮬레이션 결과이다. perVC-Queueing with threshold 방식의 경우 MCR이 2 Mbps, 4 Mbps인 VC들이 이상적인 TCP goodput보다 높은 수율을 나타냄으로써 낮은 MCR의 VC가 높은 MCR보다 성능이 좋은 결과를 보였다. 반면 제안된 알고리즘의 경우 이상적인 TCP goodput에 근접하여 모든 VC의 MCR를 충족하고 공평성도 perVC-Queueing with threshold 보다 좋은 결과를 보였다.

표 2. 제안된 알고리즘과 perVC-Queueing with threshold의 성능 비교 (LBO = 1K셀)

Performance	perVC-Queueing with proposed	perVC-Queueing with threshold
Fairness index	0.99704119	0.9758511
Total goodput	46.66 Mbps	46.49 Mbps

표 2는 전체 TCP goodput과 공평성 지표이다.

표 3. LBO에 따른 성능(Mbps)

	LBO	500	1000	1500	2000	3000
perVC-Queueing with threshold	total goodput	45.14	46.49	46.72	46.74	46.75
	fairness	0.963026	0.975851	0.9952231	0.9996278	0.9997054
perVC-Queueing with Proposed	total goodput	46.06	46.66	46.74	46.75	46.75
	fairness	0.998233	0.997041	0.9981	0.9997623	0.9998141

제안된 알고리즘의 경우 TCP goodput에서 0.17 Mbps 개선되었고 공평성 지표는 0.99704119로 perVC-Queueing with threshold 방식보다 0.023 정도 개선되었다.

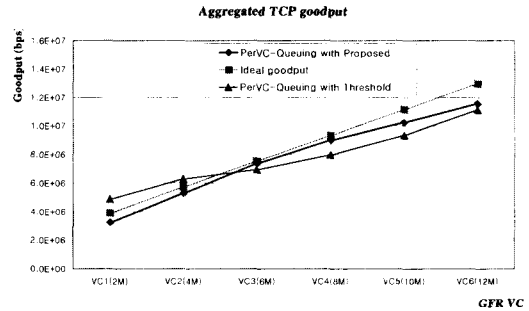


그림 5. TCP goodput(LBO = 1K 셀)

4.1 LBO의 영향

perVC-Queueing 방식에서 LBO 설정이 성능에 미치는 영향을 알아보기 위하여 LBO를 500 셀부터 3K 셀까지 변화시켜 시뮬레이션을 수행하였다. 그림 6은 LBO에 따른 공평성 지표로 제안된 알고리즘의 경우 LBO에 관계없이 좋은 공평성 지표를 유지하는 반면 perVC-Queueing with threshold의 경우 LBO가 낮을 때 상당히 낮은 공평성 지표를 나타낸다.

표 3은 제안된 알고리즘과 perVC-Queueing with threshold 알고리즘을 적용했을 경우, LBO에 따른 전체 수율과 공평성 지표 수치를 보여준다. LBO가 증가할수록 공평성과 수율이 증가하였다.

4.2 Parking-Lot 모델

TCP 연결간의 RTT와 MSS 파라미터가 TCP의 성능에 미치는 영향을 평가하기 위해서 그림 7과 같은 Parking Lot 네트워크 모델을 사용하여 시뮬레이션을 수행하였다. VC1 ~ VC3는 SW1과 SW2

를 거쳐 VC1*~VC3*와 연결되므로 링크 지연시간은 8 msec가 된다. 반면 VC4~VC5 경우 SW1, SW2와 SW3를 거치므로 링크 지연시간은 총 15 msec가 된다. 제안된 알고리즘의 성능을 perVC-Queuing with threshold와 FIFO 기반의 Double-EPD, DFBA와 비교하였다.

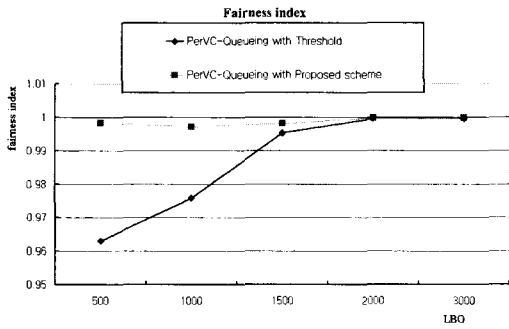


그림 6. LBO에 따른 전체 TCP goodput

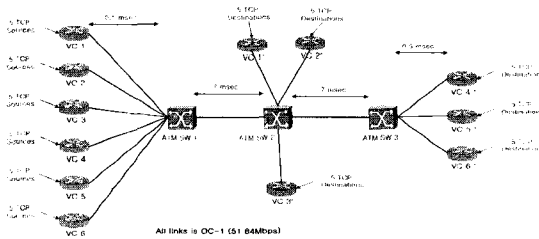


그림 7. Parking lot 시뮬레이션 모델

4.2.1 지연시간이 성능에 미치는 영향

지연시간이 알고리즘에 따라 TCP 성능에 미치는 영향을 실험하기 위해 VC1 ~ VC6의 MCR을 각각 3, 6, 9, 3, 6, 9 Mbps로 설정하고 모든 VC의 PCR은 51.84 Mbps로 설정하고 시뮬레이션을 수행하였다. 그림 8은 각 VC의 TCP goodput의 총합을 보여준다. 점선으로 표시된 그래프는 지연시간이 8 msec인 VC의 TCP goodput이고 실선으로 표시된 그래프는 지연시간이 15 msec인 VC의 TCP

goodput이다. Double-EPD와 DFBA의 경우에 같은 MCR을 가진 GFR VC를 비교했을 때 지연시간이 짧은 첫 번째 그룹의 TCP goodput이 지연시간이 큰 두 번째 그룹보다 월등히 높았다. perVC-Queuing의 경우 DFBA와 Double-EPD보다는 공평성이 우수하지만, peer to peer 모델에서의 시뮬레이션 결과보다 공평성에서 성능이 상당히 저하됐음을 알수 있다.

반면에 제안된 알고리즘의 경우에 지연시간에 관계없이 이상적인 수율에 근접해서 GFR VC가 요구하는 MCR과 공평성을 제공하였다. 표 4은 전체 TCP goodput과 공평성을 수치로 표현한 것으로 제안된 알고리즘이 Double-EPD와 DFBA보다 전체 TCP goodput과 공평성에서 우수한 성능을 보인다. 버퍼관리 알고리즘에 서비스율에 따라 패킷을 폐기하는 방식을 적용하면 TCP의 지연시간이 성능에 미치는 영향을 감소하며 네트워크 자원을 공평히 분배할 수 있다.

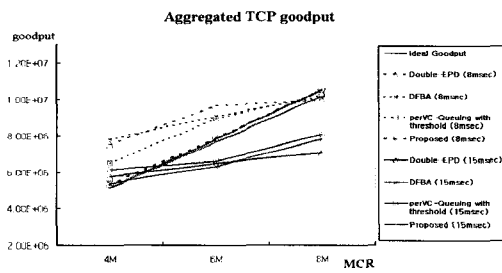


그림 8. 지연시간이 다른 환경에서 TCP goodput

4.2.2 MSS와 지연시간에 의한 영향

제안된 알고리즘과 기존의 버퍼관리 알고리즘을 지연시간과 MSS가 다를 경우, TCP의 성능에 미치는 영향을 분석하기 위해서 VC1 ~ VC6의 MCR과 PCR을 6 Mbps, 51.84 Mbps로 통일하고, MSS를 표 5와 같이 설정하여 시뮬레이션을 수행하였다.

표 4. 지연시간이 다른 환경에서 성능비교

	Double-EPD(FIFO)	DFBA(FIFO)	perVC-Queuing with threshold	Proposed
Total TCP goodput	46.46 Mbps	46.58 Mbps	46.55 Mbps	46.73 Mbps
Fairness Index	0.9395	0.9412	0.9685	0.9999

표 5. VC 파라미터

VC	MSS (bytes)	link delay (msec)
VC1	960	8
VC2	3,480	8
VC3	7,680	8
VC4	960	15
VC5	3,480	15
VC6	7,680	15

각 버퍼관리 알고리즘에 따른 시뮬레이션 결과가 그림 9과 표 6에서 보여준다. 그림 9의 각 VC의 TCP goodput의 그래프를 고찰하면, 지연이 같은 VC라도 MSS가 클수록 높은 TCP goodput을 보인다. 같은 MSS인 경우에는 지연이 작은 VC가 높은 TCP goodput을 얻을 수 있다. 표 6은 알고리즘에 따른 전체 TCP goodput과 공평성지표를 보여준다. Double-EPD와 DFBA의 경우, 공평성에서 0.8679와 0.8576의 낮은 공평성을 보였으며, perVC-Queuing with threshold의 경우도 0.9495로 지연시간이 다른 경우의 결과보다 성능이 저하되었다. 제안된 알고리즘의 경우 각 VC의 goodput이 이상적인 goodput에 근접하여 좋은 공평성을 보였다.

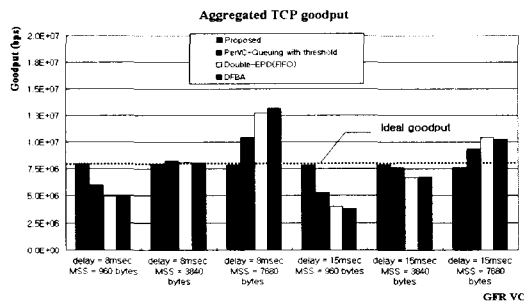


그림 9. 지연시간과 MSS에 따른 TCP goodput

표 6. 지연시간과 MSS에 따른 성능비교

	Double-EPD(FIFO)	DFBA(FIFO)	perVC-Queuing with threshold	Proposed
Total goodput	46.84 Mbps	46.83 Mbps	46.88 Mbps	46.86 Mbps
Fairness Index	0.8679	0.8576	0.9495	0.9997

주요 원인은 TCP의 congestion avoidance 알고리즘이 한 RTT동안 congestion window의 크기를 1 세그먼트씩 증가시키는데 이것은 세그먼트의 크기가 큰 TCP 연결이 한 RTT 동안 보내는 데이터의 양이 세그먼트가 작은 TCP 연결보다 많은 것을 의미한다. 제안된 알고리즘의 경우 각 VC의 평균 서비스율을 측정하여 동적 임계치에 따라 VC의 가중치보다 더 많이 서비스 받는 VC의 패킷을 선택적으로 폐기하여 GFR이 요구하는 MCR을 보장하고 높은 공평성을 제공하였다.

V. 결 론

ATM Forum에서 인터넷 트래픽과 같은 "best-effort" 트래픽을 효율적으로 서비스하기 위해 새로운 GFR 클래스를 제안했다. 본 논문에서는 GFR 서비스 클래스에서 TCP/IP 트래픽 서비스를 위한 perVC-Queuing 기반의 버퍼관리 알고리즘을 제안하였다. 제안된 알고리즘은 VC에 동적 임계치를 제공하고 VC의 평균 서비스율을 측정하여 패킷폐기를 결정한다.

폭주상황에서 QoS를 위반한 VC의 패킷을 우선 폐기하고 남은 버퍼공간을 GFR VC에 공평히 할당하여 GFR VC가 요구하는 MCR을 보장하고 높은 공평성을 제공한다. 컴퓨터 시뮬레이션을 통하여 기존의 perVC-Queuing with threshold, Double-EPD, DFBA와 성능을 평가하였다. 시뮬레이션 결과, 기존의 버퍼관리 알고리즘은 패킷 크기와 링크의 지연시간에 따라 TCP의 성능에 많은 영향을 받았다. 반면 제안된 알고리즘의 경우 패킷 크기와 지연시간의 영향을 최소화하고 기존 알고리즘보다 공평성과 네트워크 효율 측면에서 더 우수한 성능을 보였다.

참 고 문 헌

[1] The ATM Forum Technical Committee, "Draft TM 4.1 Traffic Management Specification", ATM Forum/BTD-TM-02.02, Dec. 1998.

[2] J. Andrikopoulos, A. Liakopoulos, et. "Providing Rate Guarantees for Internet Application Traffic across ATM networks," IEEE Communications surveys, vol. 2, no. 3, 1999.

[3] R. Guerin and J. Heinanen, "UBR+ Service Category Definition," ATM Forum contribution, ATM96-1598, Dec. 1996.

[4] J. Heinanen and K. Kiliki, "A Fair Buffer Allocation Scheme," Computer Communication 21, pp 220-226, 1998.

[5] R. Goyal, R. Jain, S. Fahmy and B. Vandalore, "Buffer Management for the GFR Service," ATM Forum/98-0405.

[6] M. Katevenis et al., "Weighted Round Robin Cell Multiplexing in a General-Purpose ATM Switch Chip," IEEE JSAC. vol. 9 no. 8 oct. 1991

[7] J. Huang, B-J. Lee, and S. Khorsandi, "A Simulation study of GFR Implementation," ATM Forum contribution, ATM97-1035, Dec. 1997.

[8] O. Bonaventure. "Providing Bandwidth Guarantees to Internetwork Traffic in ATM Networks," Proc. of IEEE ATM98 workshop, May 1998.

[9] Bonaventure, Olivier, "Integration of ATM under TCP/IP to Provide Services with Guaranteed Minimum Bandwidth," Ph.D. Thesis, Université de Liège, March 1999.

[10] Goyal, R. Jain, R. Fahmy, S. Vandalore, B. Kalyanaraman, S. "Design issues for providing minimum rate guarantees to the ATM unspecified bit rate service," ATM Workshop Proceedings, IEEE, pp. 169-175, 1998.

[11] W.Stevens. "TCP slow start, congestion avoidance, fast retransmit and fast recovery algorithms", Internet RFC 2001, Jan. 1997.

[12] Rohit Goyal, Raj Jain, et. al "Improving the performance of TCP over the ATM-UBR service", Journal of Computer Communications Vol. 21, No. 10, July 1998.

김관웅(Kwan-woong Kim)

정회원



1996년 2월 : 전북대학교 전자공학과 졸업

1998년 2월 : 전북대학교 전자공학과 석사

2002년 8월 : 전북대학교 전자공학과 박사

2002년 9월 ~ 현재 : 한국표준

과학연구원 Post-Doc. 연구원

<관심분야> 컴퓨터 네트워크, 인터넷 QoS, IP over ATM

이창기(Chang-ki Lee)

정회원

1988년 2월 : 전북대학교 전자공학과 학사

1990년 2월 : 전북대학교 대학원 전자공학 석사

1994년 2월 : 전북대학교 대학원 전자공학 박사

1995년~현재 : 서남대학교 컴퓨터정보통신학과 조교수

<관심분야> VLSI 설계, 컴퓨터 네트워크