

논문 01-01-11

ATM 교환기에서 멀티미디어 트래픽 지원을 위한 효율적인 셀 큐잉 및 스케줄링 알고리즘에 관한 연구

A Study on Efficient Cell Queueing and Scheduling Algorithms for Multimedia Support in ATM Switches

朴 振秀*, 李 晟遠**, 金 榮範***

(Jin-Su Park*, Sung-Won Lee**, Young-Beom Kim***)

요 약

본 논문에서는 공유 메모리형 ATM 스위치 설계에 있어서 스위치 자원의 이용률 향상과 서비스 품질 기능 지원을 위한 버퍼관리방안을 고찰하고 여러 기법들의 성능을 비교·분석하였다. 정적 임계법(ST)와 푸시 아웃(PO) 그리고, 동적 임계법(DT)의 성능을 시뮬레이션을 통하여 비교·분석하였고, 특히 동적임계법이 트래픽 부하 및 버스티니스 (Burstyness), 복수개 출력포트간 부하의 불균형성 (Non-uniformity)등의 트래픽 특성 변화에 대해 푸시 아웃 (Pushout)에 가까운 견고성 (Robustness)을 가짐을 보였다. 또한, 서비스 품질 기능 구현에 있어서 연결승인제어 (CAC)로부터 구한 트래픽 기술자를 이용하여 각 셀 스트림의 서비스 요구조건에 맞도록 메모리 공간을 할당하는 가상분할법 (VP: Virtual partitioning)과 동적분할법 (DP: Dynamic partitioning)등의 버퍼관리 메커니즘을 고찰하였다. 이 기법들을 사용할 경우 규정된 트래픽 (Regulated traffic)과 규정되지 않은 (Unregulated) best-effort 트래픽의 공존이 가능하며 규정되지 않은 트래픽이 존재하더라도 규정된 트래픽이 연결승인제어에 의해 계산된 셀 유실률을 보장받게 됨을 컴퓨터 시뮬레이션을 통하여 보였으며, 특히 과부하 상태에서 DP가 VP에 비해 서비스품질 지원 기능 면에서 우수함을 보였다.

Abstract

In this paper, we investigated several buffer management schemes for the design of shared-memory type ATM switches, which can enhance the utilization of switch resources and can support quality-of-service (QoS) functionalities. Our results show that dynamic threshold (DT) scheme demonstrate a moderate degree of robustness close to pushout(PO) scheme, which is known to be impractical in the perspective of hardware implementation, under various traffic conditions such as traffic loads, burstyness of incoming traffic, and load non-uniformity across output ports. Next, we considered buffer management strategies to support QoS functions, which utilize parameter values obtained via connection admission control (CAC) procedures to set the threshold values. Through simulations, we showed that the buffer management schemes adopted behave well in the sense that they can protect regulated traffic from unregulated cell traffic in allocating buffer space. In particular, it was observed that dynamic partitioning is superior in terms of QoS support than virtual partitioning.

Keyword : TTL-to-CMOS, Input buffer, Activation clock pulse, SRAM, Operating power

* 三星電子 Data 共通 Lab

(Data Common Tech. Lab., Samsung Electronics)

** 韓國通信 通信網 管理센터

(Network Management Center, Korea Telecom)

接受日: 2001年 5月10日, 修正完了日: 2001년 7月15日

*** 建國大學校 電子工學部

(Dept. of Electronics Eng., Konkuk Univ.)

* 본 논문은 98년도 한국학술진흥재단의 연구비에 의해 연구되었음(KRF1998-003-E00329)

1. 서 론

현재 네트워크에 유입되는 트래픽은 인터넷 사용자의 대폭적인 증가와 멀티미디어와 같은 다양한 네트워크 애플리케이션의 등장으로 인하여 양적, 질적면에서 상당한 증가를 보이고 있으며 이를 네트워크내에서 효율적으로 처리하기 위한 방안들이 활발히 연구되고 있다. 네트워크를 전송링크와 망노드의 조합으로 봤을 때 전송링크는 광대역을 갖는 광섬유 케이블로 대체됨에 따라 문제가 상대적으로 줄어들고 있으나 망노드에 해당하는 교환기와 라우터의 경우 꾸준히 증가하고 있는 트래픽을 처리하는 데 있어서 상대적으로 고도의 트래픽 처리 기술이 요구되고 있다[10].

대량의 멀티미디어 트래픽을 효율적이고 고속으로 처리하기 위한 방안으로써 고도의 패킷처리 기능들이 점차 하드웨어화 되고 있으며 이러한 추세 중심으로 ATM (Asynchronous Transfer Mode) 기술과 이를 기반으로 한 MPLS 스위치(MultiProtocol Label Switching)등의 고속 라우터 기술을 들 수 있다. 따라서 대용량 고속 ATM 스위치 개발에 많은 연구가 진행되어 왔으며 많은 방법들이 제시되고 있다. ATM 스위치 설계를 둘러싸고 있는 문제중에 최근 가장 많은 관심을 모으고 있는 문제중의 하나는 다양한 특성의 트래픽을 특성에 맞게 처리할 수 있는 스위치내 서비스 품질 (QoS: Quality of Service) 기능 구현과 대용량 트래픽의 고속처리 기능 등을 들 수 있다 [11,12].

본 논문에서는 ATM 스위치 설계 방법중 높은 처리율(throughput)을 가지며 동일한 크기의 메모리에 대해 셀 유실률이 낮은 것으로 알려진[1] 공유 메모리형 ATM 스위치 (Shared Memory ATM Switch) 설계와 관련한 효율적인 버퍼 관리기법에 대하여 고찰하였다. 공유 메모리형 스위치는 기타 다른 형태의 ATM 스위치에 견주어 대체로 다음의 장점을 가지고 있다 [6]. 첫째, 입력 버퍼형 스위치가 갖는 생기는 HOL Blocking(Head of Line Blocking)현상이 없으므로[1] 출력 버퍼형 스위치와 동일한 성능을 갖는다. 둘째, 셀 스트림들에 대한 우선 순위(priority) 제어가 비교적 용이하기 때문에 서비스품질 기능 구현에 적합하다. 마지막으로, 멀티캐스팅(multicasting)과 브로드캐스팅(broadcasting)의 구현이 용이하다. 공유 메모리형 스위치는 이러한 장점들을 지니고 있으나 복수개의 출력 포트 들이 하나의 메모리를 공유하기 때문에 적절한

버퍼관리 기능이 없는 경우 어느 특정한 출력포트 또는 가상연결이 메모리 공간을 과도하게 점유하게 되면 처리율면에서의 성능 저하를 가져온다. 또한 복수개의 가상 연결들이 하나의 메모리공간을 사용하게 되므로 각 가상연결들의 서비스 품질에 상응하는 서비스 품질 구현 측면에 있어서 버퍼관리 기능은 매우 중요하다.

메모리를 공유하는 출력포트들간에 버퍼공간을 공정하게 분배하는 동시에 버퍼 이용률을 향상시키고자 하는 기존의 버퍼 관리 방법으로는 정적 임계법 (ST : Static Threshold), 푸시아웃 (PO: PushOut)을 들 수 있다. 정적임계법의 경우 하드웨어 구현은 매우 간단하나 공유 메모리 제어 임계치의 최적값을 결정해야 하고 교환기에 유입하는 트래픽의 특성이 시간적으로 바뀔 때마다 임계치의 최적값이 변하게 되므로 이를 매번 재설정해야 하는 단점이 있다. 푸시아웃 방법은 가장 최적의 성능을 가지는 이상적인 방법이나 기존 메모리내에 있는 모든 셀에 대한 상태정보를 유지·검색해야 하기 때문에 하드웨어 구현 면에 있어서 실용성이 현저히 떨어진다[8]. 따라서, 이 두 가지 방법이 갖는 단점을 극복하기 위한 버퍼관리 기법으로써 동적 임계법 (DT : Dynamic Threshold)[2,9]이 최근에 제안되었다.

각기 다른 서비스품질 요구조건을 갖는 셀 스트림들이 하나의 메모리를 공유하게 되면 첫째로 서비스 품질 요구조건에 부합하도록 논리적인 공유 메모리의 분할 할당이 필요하며 둘째로 연결설정단계에서 정해진 트래픽 협정에 순응하는 셀 스트림(regulated traffic)을 그렇지 못한 셀 스트림 (unregulated traffic) 으로부터 보호할 필요성이 생긴다. 그러나 공유 메모리의 논리적 분할에 있어서 수반되는 여러 패러미터들의 값을 어떻게 정하느냐 하는 문제가 발생한다.

본 논문에서는 공유 메모리형 ATM 스위치 설계에 있어서 스위치 자원의 이용률 향상과 서비스 품질 기능 지원을 위한 버퍼관리방안을 고찰하고 여러 기법들의 성능을 비교·분석하였다. 첫째, 정적 임계법 (ST)와 푸시 아웃(PO) 그리고, 동적 임계법(DT)의 성능을 시뮬레이션을 통하여 비교·분석하였고, 특히 유입 트래픽의 특성 변화에 따라 임계치를 동적으로 조정하는 동적임계법이 트래픽 부하 및 버스티니스 (Burstyness), 복수개 출력포트간 부하의 불균형성 (Non-uniformity)등의 동적 트래픽 특성 변화에 대해 하드웨어 구현이 사실상 비현실적인 버퍼관리 기법인 푸시아웃에 가까운 견고성 (Robustness)을 가짐을 보

였다.

두 번째로, 서비스 품질 기능 구현 측면에서 다른 서비스품질을 가지는 셀 스트림에 대해 차등적인 서비스를 하기 위한 목적으로서 공유 메모리를 논리적으로 분할할 경우 파라미터 값들의 결정에 있어서 연결승인제어 단계에서 구한 트래픽 기술자를 사용하는 가상분할법 (VP: Virtual partitioning)[4,5]과 동적분할법 (DP: Dynamic partitioning)[3]등의 버퍼할당 메커니즘의 성능을 비교하였다. 시뮬레이션을 통하여 순용 트래픽과 비순용(Unregulated) best-effort 트래픽이 하나의 메모리를 공유하더라도 순용 트래픽의 셀 유실률이 보장됨을 보였으며, 과부하 상태에서 DP가 VP에 비해 비순용 트래픽에 의한 서비스품질 저하 측면에서 순용 트래픽을 비순용 트래픽으로부터 더 잘 보호할 수 있음을 보였다.

본 논문의 구성은 다음과 같다. 다음 장에서는 공유 메모리형 스위치를 포함한 일반적인 ATM 스위치의 구조와 기존의 버퍼관리기법들을 소개하고 동일한 트래픽을 공유 메모리 스위치에서 효율적으로 사용할 수 있는 기법인 동적임계법에 대해 고찰한다. 3장에서는 연결승인제어 (CAC)와 다른 셀 손실 조건을 가지는 트래픽을 CAC 파라미터를 이용해서 효율적으로 제어할 수 있는 기법인 VP와 DP에 대하여 기술한다. 제4장에서는 각 버퍼 관리 기법의 성능을 컴퓨터 모의 실험을 통해 비교하고 그 결과를 분석하며 마지막으로 제5장에서 결론을 기술한다.

II. 공유메모리형 ATM 스위치의 버퍼 관리 기법

공유 메모리형 스위치는 그림 1에서 보인 것과 같이 모든 입/출력 단자들이 하나의 버퍼 메모리를 공유하며 이를 교환의 방법으로 이용한다.

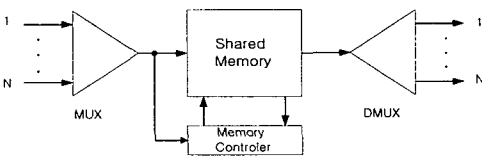


그림 1. 공유 메모리 스위치의 구조

Fig. 1. The architecture of shared memory ATM switches

버퍼 관리 기법을 설계함에 있어서 중요한 문제중의 하나는 각 가상 큐들의 크기를 제한함으로써 버퍼 할당을 보장하는 한편 전체 버퍼 공간의 공유를 통한 공유 메모리 공간의 이용률을 최대화하는 데 있다. 그러나 이 두 가지 목표는 서로 상충되는 것으로서 서로 상반되는 문제점들에 대한 절충점을 찾아내는 것이 매우 중요하다. 이 장에서는 동일한 트래픽에 대해서 공유 메모리를 공정하고, 효율적으로 사용하는 방법들에 대해서 알아본다.

2.1 정적 임계법과 푸시아웃

현재까지 공유 메모리 제어에 있어서 대표적인 방법은 정적임계법과 푸시아아웃의 두 가지 방법을 들 수 있다. 각 방안을 간단히 설명하면 다음과 같다. 정적 임계법 (ST : Static Threshold)에서는 각 가상 큐의 크기를 고정된 임계값으로 제한하는 방법을 취하고 있다. 이 방법에서는 각 출력포트에서 큐의 길이가 주어진 임계값보다 작을 경우에만 도착하는 패킷의 유입을 허용하게 된다. 푸시아아웃 (PO : PushOut) 방법에서는 도착하는 셀들을 버퍼가 가득 찰 때까지는 받아들이며 일단 버퍼가 차게 되면 이후에 유입하는 셀들은 버퍼에 들어 있던 기존 셀들 중 하나를 선택하여 그 자리에서 뺏어쓰는 방법이다.

정적임계법의 경우 하드웨어 구현은 매우 간단하나 공유 메모리 제어 임계치의 최적값을 결정해야 하고 교환기에 유입하는 트래픽의 특성이 시간적으로 바뀔 때마다 임계치의 최적값이 변하게 되므로 이를 매번 재설정해야 하는 단점이 있다. 푸시아아웃 방법은 가장 최적의 성능을 가지는 이상적인 방법이나 기존 메모리내에 있는 모든 셀에 대한 상태정보를 유지·검색해야 하기 때문에 하드웨어 구현 면에 있어서 실용성이 현저히 떨어진다. 따라서, 이 두가지 방법이 갖는 단점을 극복하기 위한 버퍼관리 기법이 필요하다.

2.2 동적 임계법 (Dynamic Threshold Scheme)[3]

이 절에서는 참고문헌[3]에서 제안한 동적임계법을 간략히 소개하고 정상상태에서 버퍼에 사용되지 않고 남아 있는 메모리의 크기를 계산하며 동적임계법의 성능에 미치는 잔여 메모리의 역할을 분석하기로 한다.

모든 가상 큐들이 공유메모리를 공정하게 사용할

수 있도록 가상 큐의 길이를 통제하는 데 사용되는 임계치를 제어 임계치 (control threshold)라 한다. 동적 임계법에 있어서 제어 임계치의 크기는 현재 사용되지 않은 전체 버퍼 공간의 크기에 의해 결정된다. 각 출력포트의 큐는 사용되고 남은 버퍼를 임의의 함수 f 에 대입하여 큐의 길이를 제한한다. 임의의 시간 t 에서, $T(t)$ 를 시간 t 일 때의 제어 임계치라 하고, $Q^i(t)$ 를 큐 i 의 길이라 정의하자. 모든 가상 큐들의 합을 $Q(t)$ 라 하면, $Q(t)$ 는 전체 공유 메모리를 차지하는 셀의 개수를 나타내게 된다. 전체 버퍼 공간을 B 라 하면, 제어 임계치 $T(t)$ 는 다음 식으로 주어진다.

$$T(t) = f(B - Q(t)) = f\left(B - \sum_i Q^i(t)\right) \quad (1)$$

만약, 시간 t 에서 $Q^i \geq T(t)$ 이면, 가상 큐 i 에 도착하는 셀은 차단된다. 이 경우 이 가상 큐로 향하는 모든 셀들은 그 큐의 길이가 제어 임계치보다 작거나 임계값이 커질 때까지 차단된다. 제어 임계치를 구하는 가장 간단한 방법은 사용되고 남은 버퍼에 α 배 곱한 값을 사용하는 방법을 고려할 수 있으며 이를 다음 식으로 나타낼 수 있다.

$$T(t) = \alpha \cdot (B - Q(t)) = \alpha \cdot \left(B - \sum_i Q^i(t)\right) \quad (2)$$

정상상태에서 $Q(t)$ 의 값을 Q , $T(t)$ 의 값을 T , S 를 활동 상태 큐의 수라고 정의하면 정상상태에서 전체 버퍼 사용공간 Q 는 다음 식으로 주어진다.

$$Q = S \cdot T + \Omega \quad (3)$$

여기서 Ω 는 제어되지 않은 큐들에 의해 점유된 공간이다. 즉, 제어 임계치보다 낮은 길이의 큐가 차지하고 있는 공간이다. 정상상태에서 각 제어된 큐의 길이는 식(3)을 식(2)에 대입하여 T 에 대해서 풀면 다음의 식을 얻는다.

$$Q^i = T = \frac{\alpha \cdot (B - \Omega)}{1 + \alpha \cdot S} \quad (4)$$

이 알고리즘에 의해서 남겨진 메모리의 크기 Θ 는

다음과 같다.

$$\Theta = \frac{B - \Omega}{1 + \alpha \cdot S} \quad (5)$$

DT에 남겨진 버퍼는 과도상태(transient state)동안 완충 공간의 역할을 하게 되어 과도상태 동안 셀의 손실을 줄이게 된다. 그리고, 출력포트 큐에서 부하가 증가될 때 남겨진 버퍼에 셀들이 유입되어 부하 조건이 바뀌면 임계값의 조정이 필요하게 된다. 식 (5)에서 α 값이 큰 값이라면, 정상상태에서 할당되지 않은 메모리 Θ 는 무시될 수 있다. 너무 큰 값으로 α 값이 주어진다면, 비록 버퍼 이용률면에서는 바람직해도, 임계값 $T(t)$ 가 너무 커지게 되어 메모리 제어가 비효율적이 된다. 즉, 하나의 가상 큐가 많은 메모리를 장시간 차지하게 되면, 다른 가상 큐들의 메모리가 부족하게 되어 셀 유실률이 커지게 된다.

III. 동적 분할법

지금까지 동일한 손실 조건을 가지는 트래픽에 대해서 효율적으로 공유자원을 사용하는 버퍼 관리 기법들에 대해 살펴보았다. 본 논문에서 제시될 기법으로는 가상 분할법(VP : Virtual Partitioning)과 동적 분할법(DP : Dynamic Partitioning)이 있으며, 이 기법들은 CAC로부터의 트래픽 묘사자를 이용하여 버퍼를 할당한다. 여기에서 논의된 방법은 구현의 제약과 구현의 용이성 때문에 다음과 같은 조건이 필요하다.

첫째, 차단 방법(blocking scheme)이 되어야 한다. 즉, 단일 큐에 대하여 사용할 수 있는 버퍼의 양은 제한되어야 하고, 제한이 침해되었을 경우는 큐에 도착하는 패킷들이 차단되어야 한다. (이와 반대되는 것이 Pushout 방법이다.)

둘째, 시간의 영향을 받지 않는 방법(time-insensitive scheme)이 되어야 한다. 즉, 매 도착에 대하여 승낙과 거절에 대한 결정이 이전의 결정들에 대하여 독립적으로 수행되어야 한다. 전체 상태는 단지 시스템에서 큐의 길이와 전체 점유로 구성되어 있다.

셋째, 쉽게 계산되어 질 수 있는 파라미터들로 되어야하고 동작 원리가 간단해야 한다.

3.1 가상 분할법(Virtual Partitioning)[5]

최근에 제안된 가상 분할(VP : Virtual Partitioning) 개념을 이용하여 BAC (Buffer Admission Control)가 기반이 되는 버퍼 관리 기법으로서, 연결 승인 제어(CAC: Call Admission Control)가 적용이 되어 공정하고 효율적으로 자원을 사용한다. VP는 규정된 트래픽과 규정되지 않은(best-effort)트래픽의 공존을 허용하여 공유 버퍼 자원에서 다중화. 이득을 얻을 수 있을 뿐만 아니라 차등적인 할당을 제공하여 커넥션의 보호와 격리를 할 수 있다. 이 기법에서, 각 가상 큐 i 는 CAC로부터 버퍼 용량 B_i 를 할당받는다.

$$\sum_i B_i \geq B_f \tag{6}$$

여기서, B_f 는 사용 가능한 버퍼의 크기이다.

만약, 큐의 길이 $Q_i(t)$ 가 B_i 보다 작으면, 그 큐의 상태는 저부하(underload)라고 정의하고, 그 반대의 경우를 과부하(overload)라고 정의한다. 두 개의 예약 파라미터 R_u 와 R_o 는 각각 저부하 상태와 과부하 상태에 대해 정의된다 ($0 \leq R_u \leq R_o \leq B_f$).

다음 조건 중 하나만 만족되면 도착되는 셀은 받아들여진다.

$$\begin{aligned} Q_i < B_i \text{ and } Q(t) < B_f - R_u \\ Q_i \geq B_i \text{ and } Q(t) < B_f - R_o \end{aligned} \tag{7}$$

여기서 $Q(t)$ 는 전체 큐 길이이다.

예약 파라미터 (R_u, R_o)를 잘 선택하면 이 방법으로 높은 수준의 처리율을 얻을 수 있고, 공유메모리 시스템에서 차등적인 할당을 제공하는 CAC와 관련되어 사용될 수 있다. 이 방법에서의 한계는 각 큐들의 길이를 직접적으로 제어하지 못한다는 점이다. 더군다나 이 방법은 연결 당 할당의 단일 값을 사용하기 때문에, 이것은 때때로 다른 손실 정도를 제공하는 데 어려움 뿐 아니라 규정되지 않은 소스들이 시스템에 승인되는 것을 과도하게 제한하지 않으면서 규정되지 않은 트래픽과 규정을 순응하지 않는 소스들로부터 보호가 어렵다.

3.2 동적 분할법 (Dynamic Partitioning)

본 논문에서 사용한 스위치 모델은 그림 2에서와 같이, 각 출력 링크로 향하는 몇 개의 연결들(또는 flow나 소스)을 가지고 있는 $N \times N$ 공유 메모리 ATM 스위치이다. 큐잉 구조는 단일 큐 당 출력포트로 구성되어 있거나, 연결 당 각 큐들로 구성되어 있다. 이들 중 어떤 경우라도 적당한 스케줄링 방법이 요구된다. 예를 들어, 출력포트 당 하나의 큐라면, 간단한 first-in-first-out (FIFO)이 적당하며, 반면 연결 당 큐들에 대하여는 Generalized Processor Sharing(GPS)이 쓰여질 수 있다.

사용되는 스케줄링 방법은 CAC에 의해 결정되는 것에 따라야 한다. 큐잉 구조와는 상관없이 각 연결 i (per-flow accounting)에 대한 큐-점유 카운터가 제공되어야 할 것이다.

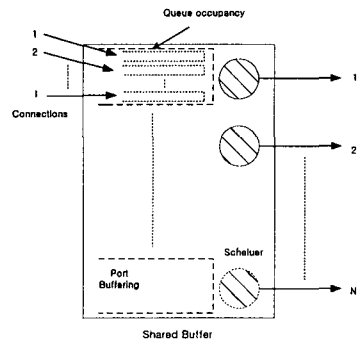


그림 2. 스위치 모델
Fig. 2. The switch model

IV. 시뮬레이션을 통한 성능 비교 및 검토

본 장에서는 먼저 모의 실험에 사용될 모델에 대해서 알아보고, 각 버퍼 관리 기법에 대한 실험 결과를 살펴본다.

4.1 모의 실험 모델

각 입력은 하나의 버스트(burst) 소스이다. 소스는 활성(active)과 비활성(idle) 기간으로 바뀌면서 활성기간에 셀을 내보내게 된다. 활성 주기 동안은 파라미

터 α 를 갖는 지오메트릭(geometric) 분포가 된다. 반면 비활성 주기 기간동안은 파라미터 β 를 갖는 지오메트릭 분포가 된다. 여기서 각 활성기간 동안 적어도 하나의 셀이 발생하고 비활성 기간에는 셀이 발생하지 않고, 버스트 길이는 통계적으로 독립이라고 가정한다.

α 와 β 가 주어졌을 때, 평균 버스트 길이를 L_b , 평균 비활성 시간을 L_{idle} , 그리고 출력 링크에 정규화된(normalized) 부하를 ρ 라 할 때 다음과 같다.

$$L_b = \frac{1}{\alpha}, \quad L_{idle} = \frac{1}{\beta}, \quad \rho = \frac{L_b}{L_{idle} + L_b} \quad (8)$$

여기에서 L_b, L_{idle} 의 단위는 시간을 1개 셀 전송시간으로 분할했을 때의 타임 슬롯수 (Number of time slots)이다. 예를 들어 100Mbps 속도의 전송라인의 경우 1 타임슬롯은

$$1 \text{ 타임슬롯} = \frac{1 \text{ cells}}{1 \text{ Mbps}} = \frac{64 \text{ bytes} \times 8 \text{ bits}}{1 \text{ Mbps}} = 512 \text{ msec}$$

으로 주어진다.

스위치에서 통과된 셀과 유실된 셀의 비율을 나타내는 셀 유실률(CLR : Cell Loss Ratio)을 성능평가 지수로 정했으며 셀 유실률은 다음의 식으로 주어진다.

$$CLR = \frac{\text{손실된 셀 갯수}}{\text{전송된 셀 갯수}}$$

4.2 동적 임계법의 성능평가

모의 실험에서 사용된 스위치 구조는 single-stage 스위치이고, 입력과 출력 링크의 속도는 같다고 가정한다. 이 스위치는 출력 버퍼형 구조를 가지며 공유 메모리를 사용한다. 따라서 스위치의 각 출력포트는 하나의 논리적 큐를 가지고 있으나, 이들 큐들은 모두 같은 스위치 메모리를 공유한다.

첫 번째 실험은 각 버퍼 관리 기법에 대한 셀 유실률을 알아보았다. 스위치의 크기는 512×512 , 전체 버퍼의 크기 B 는 6500셀이고, 처리율 ρ 는 0.55, 버스트 크기 L_b 는 20셀이다. 식 (8)을 이용하여 α 와 β 값을 계산하면, α 는 0.05, β 는 0.06이 된다. 이 경우 4가지 방법 NC(No Control), ST(Static Threshold),

PO(Push Out), DT(Dynamic Threshold)를 이용하여 CLR을 살펴보면 다음과 같다. (단, ST의 정적 임계값은 120셀로 하였다.)

표 1. 여러 가지 버퍼 관리 방법에 대한 셀 유실률
Table 1. The cell loss ratio for various buffer management methods

	NC	ST	PO	DT
CLR	0.01885	0.007466	0.002810	0.00560

표 1에서와 같이 버퍼 관리 기법을 쓴 경우가 그렇지 않은 경우보다 더 좋은 성능을 보이고 있으며 PO, DT, ST 순서로 좋은 성능을 나타냄을 알 수 있다.

두 번째 실험에서는 하나의 포트에 부하를 크게 주었을 때 나머지 포트들의 셀 유실률을 측정하였다.

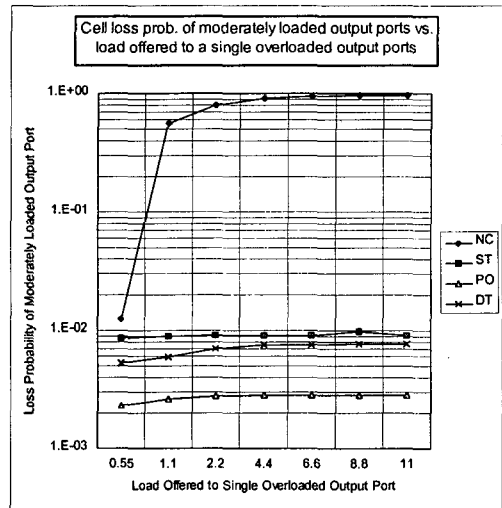


그림 3. 단일 과부하 포트에 대한 나머지 포트들의 셀 유실률

Fig. 3. The cell loss ratio of remaining ports in the presence of an overloaded port

그림 3에서와 같이 단일 과부하 포트에 대한 각 버퍼 관리 방법들의 셀 유실률은 PO, DT, ST 순으로 좋은 성능을 나타내고 있다는 것을 알 수 있다.

세 번째 실험에서는 부하 조건이 틀려질 경우에 대하여 실험을 하였다. 그림 4은 부하가 각각 0.35, 0.4, 0.5, 0.55, 0.6, 0.65, 0.7로 바뀔 때, 셀 유실률을 나타낸 것이다.

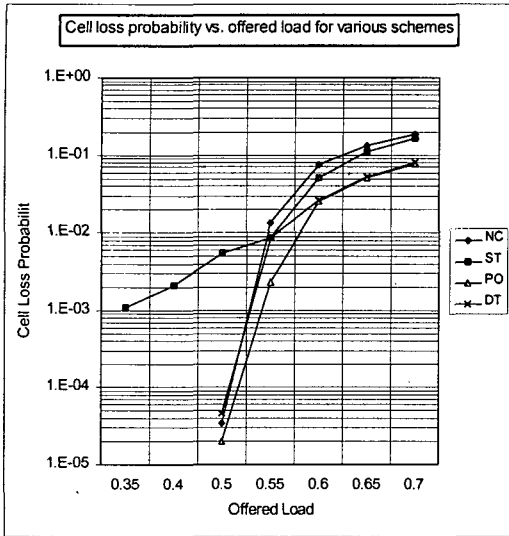


그림 4. 부하조건이 바뀔 때 셀 유실률

Fig. 4. The cell loss ratio according to traffic loads

그림 4에서 보여지듯이 역시 PO가 가장 좋은 성능을 나타내고 있으며, DT는 PO와 비슷한 성능을 보여주고 있다. 한편, ST는 부하가 0.55보다 적을 때 NC보다도 오히려 셀 유실률이 큰 것을 보여주고 있는데, 부하가 작을 때는 전체 버퍼를 공유하는 것보다 분리하는 것이 더 바람직하다는 것을 알 수 있다.

네 번째 실험은 세 번째 실험과 비슷한 목적의 실험으로서 소스의 상태를 바꾸어 가면서 실험을 하였다. 각 소스의 부하 ρ 는 0.55로 하고, 그 대신 각 소스의 버스트 크기를 변화하면서 실험을 하였다. 실험 결과는 그림 5와 같다.

그림 5에서와 같이 MBS가 커지게 되면 모든 방법에서 셀 손실이 커지게 된다는 것을 알 수 있다. 이 실험에서는 PO가 DT보다 약간 성능이 좋은 것으로 나타났다. 나머지 ST와 NC의 결과에서는 세 번째 실험과 비슷한 결과가 나왔는데, 이것은 세 번째 실험에서 설명한 내용과 유사하다고 볼 수 있다.

4.3 동적 분할법

여기에서는 셀 레벨 모의 실험으로 각 버퍼 관리 기법의 성능을 알아보고, 커넥션 레벨 모의 실험을 통해 주로 VP와 DP에서 규정된 트래픽의 셀 유실률을 대해 비교했다

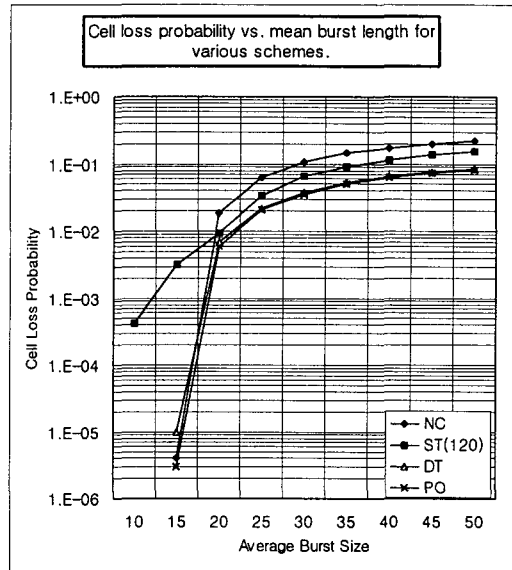


그림 5. 소스의 MSB가 변할 때 셀 유실률($\rho=5.5$)

Fig. 5. The cell loss ratio according to MSB of source stream($\rho=5.5$)

4.3.1 셀 레벨 모의 실험

스위치 모델과 성능 평가 항목은 동적임계법의 경우와 같다. 이번 모의 실험에서의 스위치는 출력포트당 논리 큐가 있는 16×16 스위치이다. 각 입력 링크에는 하나의 on-off 소스가 있고, 평균 버스트 길이는 L_0 인 지오메트릭(geometric) 분포를 갖고, 입력 링크와 출력 링크의 속도는 같다고 가정하고, 출력 링크에 정규화된(normalized) 부하는 ρ 이다. 출력포트에 대해 non-uniform한 분포를 가지며, 두 종류(class)를 가진다. 하나는 알맞은(moderately) 부하를 가진 포트이며 부하는 60% 이고, 다른 하나는 과부하(heavily) 포트이며 부하는 120%이다. 평균 버스트 길이는 100 셀이며 최대 셀 율(PCR)은 링크 용량과 같다. 전체 버퍼크기 B_j 는 1536셀 이다. 그림 6은 임계값 T의 변화에 따른 ST의 성능과 VP, DP의 성능을 보인 것이다. DP의 파라미터 a 는 16, $b_{0,i}$ 는 128, γ 는 1502, b_i 는 $B_j/N=96$ 이고, VP의 파라미터 B_i 는 128, 예 약 파라미터 R_w , R_o 는 각각 0, 32이다.

이 실험에서 NO는 No Control이다. 즉, 임계값 T는 B_j 가 된다. 또 ST는 정적 임계법을 뜻하며 괄호

안의 숫자는 임계값이다. 임계값이 96일 경우는 CP(Completely Partitioning)의 경우가 된다 ($T = B_f/N = 96$).

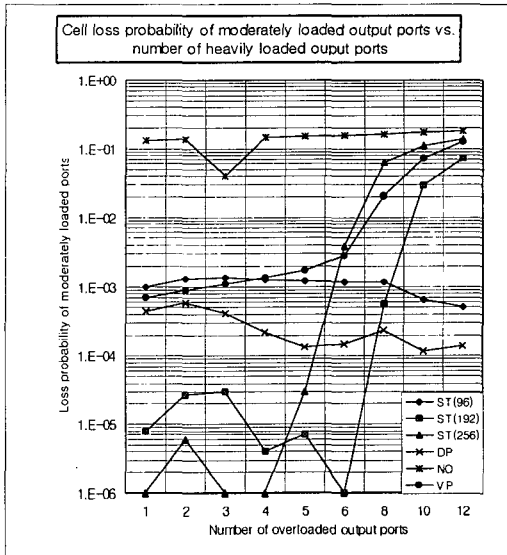


그림 6. 과부하 포트들에 대한 나머지 포트들의 셀 유실률

Fig. 6. The cell loss ratio of the remaining ports in the presence of an overloaded port

임계값이 클수록 공유효과는 커지게 된다. 과부하가 걸린 포트의 수가 적을 때는 임계값이 큰 ST가 셀 유실률이 작으며, 과부하 포트수가 많아지게 되면 임계값이 작은 ST가 셀 유실률이 작다. 이것은 부하가 많이 걸려 있는 포트들이 버퍼를 많이 차지하고 있더라도 작은 부하가 걸리는 포트들에서 계속해서 셀이 유실되기 때문이다. VP의 경우에는 과부하 포트수가 많을 때는 높은 셀 유실이 생긴다. 이것은 과부하 상태일 때 적용되는 전체 셀의 합이 $B_f - R_0$ 까지 허용되는 방식으로 인해서 각 큐들의 길이를 각각 제한을 하지 못하기 때문이다. 한편, DP에서는 항상 VP보다 더 좋은 성능을 보여주고 있다. 그 이유는 ST와 VP와는 반대로 과부하 포트들이 증가해도 각 큐들의 버퍼 공간을 보장을 해줄 수 있기 때문이다.

4.3.2 커넥션 레벨 조정 실험

여기에서는 하나의 포트에 여러 개의 커넥션 연결

을 가정하여, 비순응 연결 또는, best-effort 연결과 순응 연결이 공존할 때 각 버퍼 관리기법의 성능을 알아보겠다. 실험은 먼저 버퍼 관리 기법이 없는 경우를 살펴본 후, VP와 DP방법으로 실행하여 성능을 비교하겠다.

실험 모델은 4x4 입출력포트를 가지는 스위치이며, 전체 버퍼 크기 B_f 는 400셀이고, 4-1절의 모델과 같다. 각 커넥션마다 트래픽 모델은 정규화된 출력 링크의 10%의 부하를 가지는 지오메트릭(geometric)분포를 갖는 on-off 소스라고 가정한다. 그리고, 순응 커넥션을 만들기 위해서 각 소스를 리키 버킷 레귤레이터(leaky bucket regulator)를 통과시킨다. 리키 버킷 레귤레이터는 프로파일(r, B_T, P)로 소스를 규정한다. 여기서 프로파일 요소 r 는 토큰 속도, B_T 는 버킷 크기 P 는 최고 속도이다. 실험의 성능평가 항목은 5-2절의 CLR과 동일하며, 그림 7은 실험 모델을 도시한 것이다.

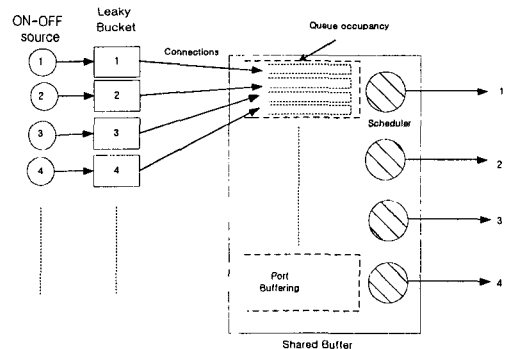


그림 7. 커넥션 레벨 실험 스위치 모델

Fig. 7. The switch model for connection level simulation

첫 번째 실험은 하나의 포트에 향하는 커넥션들이 모두 비순응 연결이고, 즉 리키 버킷이 제거된 상황일 때, 나머지 순응 연결들에 대한 버퍼 할당 보장에 관한 실험이다. 이 실험에서는 순응 연결의 버스트 크기는 120셀로 하고, 비순응 연결의 버스트 크기는 점차 늘려가며 실험을 하였다.

그림 8은 버퍼 관리 기법이 없을 경우에 규정된 트래픽과 그렇지 않은 트래픽의 셀 유실률을 규정되지 않은 소스의 버스트 크기가 증가함에 따라 나타낸 것이다. 처음 커넥션을 16개로 허가할 때의 허용 셀

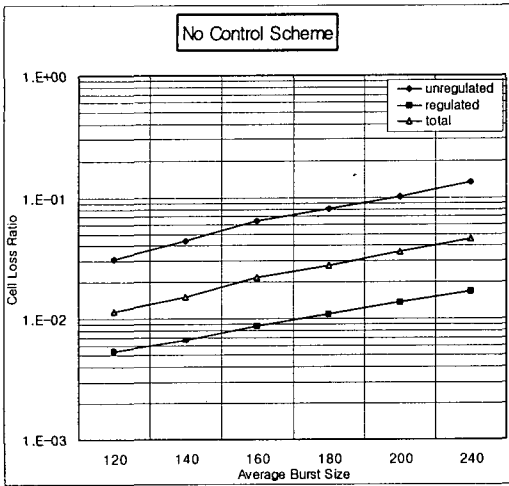


그림 8. NC에서 비순응 연결의 버스트 크기가 바뀔 때 셀 유실률

Fig. 8. The cell loss ratio according to burst sizes of unregulated connections in NC

유실률인 2×10^{-3} 을 만족하지 못하는 것을 알 수 있다.

그림 9, 10에 각각 VP와 DP에 대한 셀 유실률을 나타냈다.

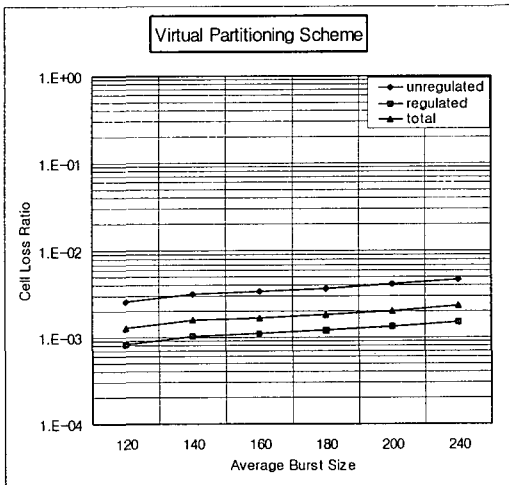


그림 9. VP에서 비순응 연결의 버스트 크기가 바뀔 때 셀 유실률

Fig. 9. The cell loss ratio according to burst sizes of unregulated connections in VP

그림 9에서와 알 수 있듯이 VP를 사용한 경우, 버퍼 관리 기법이 특별히 사용되지 않은 경우에 비하여 셀 유실률이 상당히 낮아졌을 뿐 아니라 규정된 트래픽의 순응 연결은 AC에 의해 정해진 셀 손실률을 만족한다는 것을 알 수 있다.

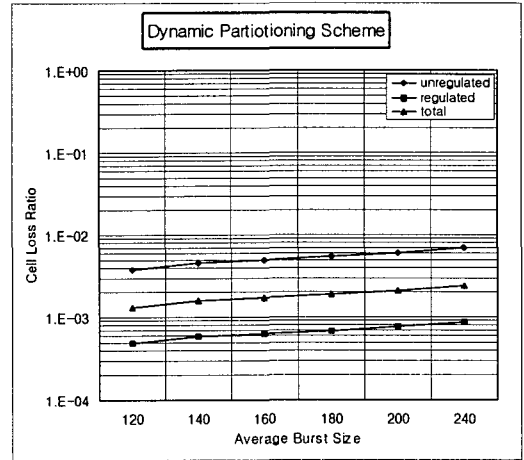


그림 10. DP에서 비순응 연결의 버스트 크기가 바뀔 때 셀 유실률

Fig. 10. The cell loss ratio according to burst sizes of unregulated connections in DP

그림 10은 DP의 셀 유실률을 나타낸 것이다. DP는 VP와 마찬가지로 순응연결에서 대해서 CAC에서 요구된 셀 유실률을 만족시킨다.

V. 결 론

현재 널리 사용되고 있는 공유 메모리형 ATM 스위치 설계에 있어서 스위치 내 공유 메모리를 효율적 사용과 출력포트간의 공정한 메모리를 할당을 위해서는 어느 특정 출력포트로 향하는 셀 트래픽이 메모리 공간을 독점하는 현상을 제어하면서 메모리 이용률을 높이기 위한 효과적인 버퍼 관리 기법이 필요하다. 또한 서비스 품질 기능 구현을 위해 각기 다른 서비스 품질을 갖는 셀 스트림에 대해 차등적으로 스위치 자원을 할당할 수 있는 자원할당 메커니즘도 요구된다. 그러나 서비스 품질 요구조건에 상응하는 서비스를 실현하는 버퍼 제어기법과 제어 파라미터의 실시간 결정 등의 문제는 매우 어려운 문제에 속한다.

본 논문에서는 먼저 정적 임계법(ST)와 푸시 아웃(PO) 그리고, 동적 임계법(DT)의 성능을 시뮬레이션을 통하여 비교·분석하였고 이 중 동적임계법이 트래픽 부하, 버스티니스 (Burstyness), 복수개 출력포트 간 부하의 불균형성 (Non-uniformity) 등의 동적 트래픽 특성 변화에 대해 하드웨어 구현이 사실상 비현실적인 버퍼관리 기법인 푸시아웃에 가까운 견고성 (Robustness)을 가짐을 보였다. 두 번째로, 서비스 품질 기능 구현 측면에서 다른 서비스품질 가지는 셀 스트림에 대해 차등적인 서비스를 하기 위해 연결승인제어로부터 구한 트래픽 기술자를 이용하여 각 셀 스트림의 서비스 요구조건에 맞도록 메모리 공간을 할당하는 가상 분할법 (VP)과 동적 분할법 (DP)을 고려하였다. 이 방법을 이용하면 규정된 트래픽과 규정되지 않은 best-effort 트래픽의 공존이 가능하며 규정되지 않은 트래픽과 함께 있을 경우 규정된 트래픽은 연결승인제어에 의해 계산된 셀 유실률을 만족한다는 것을 모의실험을 통해 확인하였다. 반면, 과부하 상태일 때, DP는 VP보다 규정된 트래픽을 규정되지 않은 트래픽으로부터 더 잘 보호할 수 있음을 보였다. 그 이유는 DP가 통계적 다중화를 이용한 버퍼 할당 메커니즘 외에도 최소 보장 할당 (Minimum guaranteed allocation) 파라미터가 함께 사용되기 때문이다.

DP는 하드웨어 구현이 간단하므로 고속 스위치에서 쉽게 적용될 수 있을 뿐만 아니라 각 트래픽의 QoS를 보장할 수 있는 버퍼 관리 기법이 될 것이다. 마지막으로 본 연구에서 고려한 성능지수는 스위치내 셀 유실률에 국한되나 스위치내에서 QoS를 보장하기 위한 요소로서 셀 전송 지연 또한 추후의 연구에서 고려될 필요가 있다고 본다.

참고문헌

- [1] F. A. Tobagi, "Fast Packet Switch Architectures for Broadband Integrated Services Digital Networks", Proc. IEEE, vol. 78, no. 1, Jan. 1990, pp. 133-67.
- [2] A. K. Choudhury and E. L. Hanhne, "Dynamic Queue Length Thresholds for Shared-Memory Packet Switches", IEEE/ACM Trans. Commun., vol. 6, no. 2, Apr. 1998, pp.130-40.
- [3] A. K. Choudhury, S. Krishnan and F. M. Chuissi, "Dynamic Partitioning: A Mechanism for Shared Memory Management", IEEE pp.144-152, 1999.
- [4] K. Kumaran and D. Mitra, "Performance and Fluid Simulations of a Novel Shared Buffer Management Scheme", Proc. IEEE INFOCOM'98, Mar. 1988.
- [5] D. Mitra and I. Ziedins, "Virtual Partitioning by Dynamic Priorities: Fair and Efficient Resource Sharing by Several Services", Broadband Communications, Proc. International Zurich Seminar on Digital Communications, pp. 173-185, Spring 1996.
- [6] Joan Garcia-Haro and Andrezej Jajszczyk "ATM Shared-Memory Switching Architectures" IEEE Network, July/August 1994.
- [7] A. Elwalid, D. Mitra and R. Wentworth, "A New Approach for Allocating Buffers and Bandwidth to Heterogeneous, Regulated Traffic in an ATM Node", IEEE Journal on Selected Areas of Communications, vol. 13, pp. 1115-1127, Aug. 1995.
- [8] M. Arpaci and J. A. Copeland, "Buffer Management for Shared-Memory ATM Switches", IEEE Communications Surveys & Tutorials, vol 3, First Quarter 2000.
- [9] A. K. Choudhury and E. L. Hahne, "A Simulation Study of Space Priorities in a Shared Memory ATM Switch", IEEE Journal High Speed Networks, vol. 3, pp. 491-512, No. 4, Nov. 1994.
- [10] A. S. Tanenbaum "Computer Networks 3rd.", Prentice Hall Inc., 1996.
- [11] W. Stallings "High Speed Networks, TCP/IP and ATM Design Principles", Prentice Hall Inc., 1998.
- [12] U. Black "ATM volume1 Foundation for Broadband Networks 2nd.", Prentice Hall Inc., 1999.

저 자 소 개

朴 振 秀 (正 會 員)

1998년 2월 건국대학교 전자정보통신공학과 졸업
2000년 8월 건국대학교 전자정보통신공학과 석사
2000년 9월 ~ 현재 삼성전자 네트워크사업부

李 晟 遠 (正會員)



1990. 2 승실대학교 전자공학과 졸업

1994. 8 승실대학교 산업대학원 전자공학과 졸업

1990. 1 ~ 1998.12 머큐리(舊 대우통신) 선임연구원

국설 및 수출형 교환기 H/W개발, 개량개선 및 인증업무수행

DELMONS (전용회선집중보전시스템) 국산화 업무 수행

1999. 2 ~ 현재 한국통신 통신망관리단 전송기술팀 재직중

2000. 3 ~ 현재 건국대학교 대학원 전자정보통신공학과

관심분야

교환 및 광전송시스템(WDM)

통신망 신뢰도 분석 및 통신망관리

金 榮 範 (正會員)



1984년 서울대학교 전자공학과 졸업

1986년 서울대학교 전자공학과 석사

1996년 미국 메릴랜드주립대 (College Park, MD) 전자공학 박사

1986년 ~ 1988년 한국통신 품

질보증단 전임연구원

1997년 ~ 현재 건국대학교 정보통신대학 전자공학부 조교수