

# ATM망에서 흐름제어 셀 다중화기

## (A Rate-Controlled Cell Multiplexer in ATM Networks)

강 구 흥 \* 박 상 조 \*

(KooHong Kang) (SangJo Park)

**요 약** 비동기식전송모드(Asynchronous Transfer Mode) 망에서 선입-선출(First Come First Served)은 각 연결들이 요구하는 다양한 서비스 품질(QoS : quality of service)을 보장하기 어렵다. 따라서 각 VC(Virtual Connection)에 확실한 대역폭 할당을 통해 요구하는 QoS를 보장하는 트래픽 분리 기법이 제안되었다. 본 논문에서는 이러한 목적을 위해 VC 단위의 “주기적 토큰-기반 조절기”를 사용하는 흐름제어 셀 다중화기를 제안한다. 제안된 다중화기는 work-conserving 성질을 유지하면서 VC 트래픽 분리를 실현하게 된다. 먼저, 제안된 다중화기의 우수한 주요 성질을 논하고 시뮬레이션을 통해 출력 트래픽 스트림 특성과 각 VC의 트래픽 분리 특징을 분석한다.

**Abstract** Since First-Come-First-Served discipline makes it difficult to satisfy the diverse quality of service(QoS) requirements for each connection, traffic isolation has been proposed for guaranteeing the desired QoS through explicit bandwidth allocation to each virtual connection(VC) in ATM networks. For this purpose, we propose a rate-controlled cell multiplexer using a simple “periodic token-based regulator” per-VC. The proposed multiplexer realizes VC isolation, while still operating work-conservingly. We first discuss its major properties, and then examine its features with respect to the characteristics of the output traffic streams and the traffic isolation of individual VC's, by simulation.

### 1. 서 론

비동기식전송모드(ATM : Asynchronous Transfer Mode) 기반 광대역 종합정보통신망을 이용한 멀티미디어 응용 서비스는 셀 손실률과 셀 지연 특성에 있어서 다른 서비스 품질(QoS : Quality-of-Service)을 요구한다. ATM 망을 형성하는 스위치 혹은 다중화기와 같은 ATM 노드가 이들 QoS를 보장하기 위해서는 이들 서비스 원(source)으로부터 입력되는 이질(heterogeneous) 트래픽 스트림을 적절히 분리 할 수 있는 기능이 요구된다. 그러나 오늘날 대부분 ATM 망에서 여전히 사용되고 있는 선입-선출(FCFS : first-come-first-served)은 트래픽 분리 능력을 가지고 있지 못해 이들 다양한 QoS를 보장하기에는 어려움이 있다. 즉 FCFS

는 입력 스트림에서 도착하는 셀 순서가 소모하는 대역폭을 결정하게 되며, 따라서 각 스트림의 QoS는 자신의 트래픽 파라미터뿐만 아니라 다른 스트림의 트래픽 파라미터에 의해 크게 영향을 받게된다[1,2].

ATM 노드는 그림 1과 같이 하나의 큐잉모델로 표현된다. ATM 노드에서 셀 스케줄러(혹은 셀 스케줄링 기법)는 출력 링크 상의 매 슬롯(slot) 마다 셀을 전송하기 위해 하나의 입력 스트림을 결정한다. 따라서 셀 스케줄러에 의해 각 VC(virtual connection)에 실질적인 대역폭이 할당되며, 이로 인해 셀 스케줄러는 각 VC 연결에 대한 QoS를 결정하게 된다[2-8].

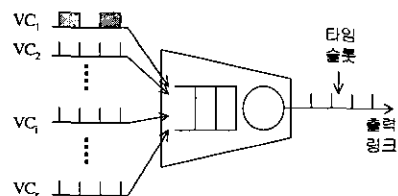


그림 1 타임 슬롯 된 ATM 노드의 큐잉 모델

\* 본 논문은 2000년도 서원대학교 융용과학연구소 연구비 지원에 의한 연구결과임.

† 정 외 원 : 서원대학교 컴퓨터정보통신공학부 교수  
khkang@seowon.ac.kr  
parks@seowon.ac.kr

논문접수 : 2000년 10월 5일

심사완료 : 2001년 6월 26일

ATM 노드의 입출력 물리 링크에 수 백개 이상의 VC가 다중화되고 ATM 노드는 고속의 셀프 라우팅(self-routing)과 넌 블로킹(nonblocking) 셀 스위칭을 수행한다. 더욱이 가까운 미래에는 광전송과 반도체 기술의 고속 성장으로 인해 ATM의 전송 및 교환의 기본 속도가 현재의 155 혹은 622 Mbps에서 수 Gbps로 증가 될 것이며, 따라서 ATM 노드는 더욱 많은 연결 수와 광속의 스위칭과 전송을 지원하게 될 것이다. 결과적으로 이러한 환경에서는 실시간 동작을 위해 하드웨어로 간단하게 실현될 수 있는 셀 스케줄링 설계가 필연적이라고 할 수 있다. 이러한 관점에서, VC 단위의 타임스탬핑(time-stamping) 혹은 임계시간 비교(deadline comparisons)[4,6]를 사용하는 셀 스케줄링 기법은 비교기 설계 및 임계시간 관리가 쉽지 않기 때문에 ATM 망에 직접 적용하기에는 어려움이 많다. 뿐만 아니라 프레임 기반 방식[7,8]은 망전체 일관성있는 시간 프레임 구조를 필요로 하기 때문에 대역폭 할당에 있어 유연성이 떨어진다.

본 논문에서는 VC 단위로 필요한 대역폭을 명확히 할당할 수 있는 흐름 제어 셀 다중화기를 제안한다. 제안된 다중화기는 망 자원을 효율적으로 사용할 수 있는 work-conserving 성질(전송할 셀이 다중화기에 존재할 때마다 출력 링크 상의 출력 슬롯이 반드시 사용되어 진다면, 해당 다중화기는 work-conserving 성질을 가진다고 하며, 그렇지 않는 경우는 non-work-conserving 성질을 가진다.)을 유지하고 있으며, 특히 각 VC는 하나의 카운터에 의해 구현되는 “주기적 토큰-기반 조절기(PTR : periodic token regulator)”에 의해 전송율이 제어됨에 따라 ATM 노드가 대형화 및 고속화되어도 하드웨어로 실현하는데 어려움이 없다. 또한 각 연결 단위로 보장되는 서비스율의 평균 구간과 실질적인 서비스 시간의 관점에서 제안된 다중화기의 우수한 주요 성질을 논한다. 시뮬레이션 결과로부터 제안된 다중화기는 출력 스트림의 버스트(burstiness) 성을 효과적으로 줄이고, VC 단위로 명확한 대역폭을 할당하는 트래픽 분리 성질을 확인할 수 있다.

2. 제안된 흐름제어 셀 다중화기

ATM 노드에서 타임-슬롯화된 출력 큐를 고려하면 그림 1과 같다. 하나의 슬롯은 하나의 ATM 셀 전송에 필요한 시간 간격이며, 이후 본 논문에서 기본적인 시간 단위로 사용한다. 이제 ATM 노드의 출력 링크 상에  $r$  개의 VC가 다중화되어 있다고 가정하고 이들 VC 집합

을  $R$ 로 표시한다. 이제  $i$ -th VC를 위해 전체 출력 링크 대역폭 중 최소 fraction  $\phi_i$  대역폭을 할당하기 위하여 다음과 같이 할당 서비스 시간(allocated service time)을 정의한다.

$$\text{할당 서비스 시간 } (D_i) = \lfloor 1 / \phi_i \rfloor$$

그림 2는 본 논문에서 제안된 PTR을 사용한 흐름제어 셀(RCC : Rate-Controlled Cell) 다중화기의 블록도다. PTR은 하나의 VC 큐(VCQ : Virtual Connection Queue)와 하나의 토큰 발생기(TG : token generator)로 구성된다. 따라서 RCC 다중화기는 복수의 PTR과 하나의 VCQ 식별자 (VID : VCQ identifier) 리스트(list), 그리고 하나의 출력 버퍼로 구성된다.  $i$ -th VC,  $i \in R$ ,로 부터 도착된 셀은 VCQ <sub>$i$</sub> 로 입력되고 토큰 발생기 TG <sub>$i$</sub> 는 그림 3에서 보여지는 바와 같이 VCQ <sub>$i$</sub> 내 셀을  $D_i$  슬롯마다 출력 버퍼로 전달한다. 결국 TG <sub>$i$</sub> 는  $i$ -th VC에 명확하게 슬롯을 할당하기 위해 매  $D_i$  슬롯마다 하나의 토큰을 발생하게 된다. 출력 버퍼는 동일한 슬롯에 여러 PTR로부터 동시에 도착하는 셀들을 일시 저장하고 FCFS 형태로 이들을 서비스하게 된다. 본 논문에서는 빈 출력 버퍼에 도착한 셀은 즉시 서비스되며, 시스템은 안정상태(stable state) (i.e.,  $\sum_{i=1}^r 1/D_i \leq 1$ )에 있다고 가정한다.

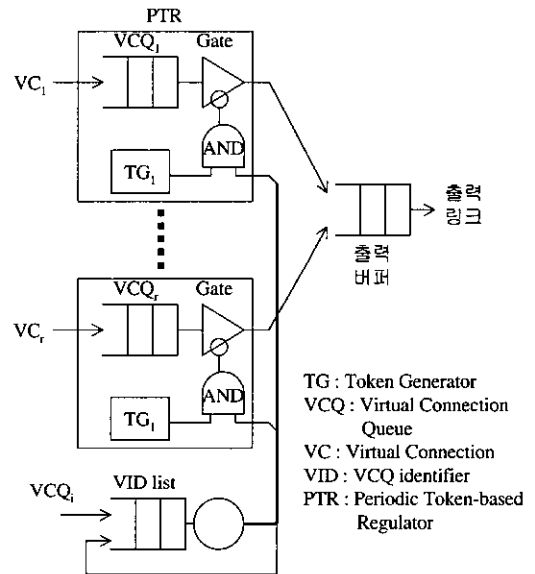


그림 2 주기적 토큰-기반 조절기(periodic token-based regulator)를 이용한 흐름제어 셀 다중화기

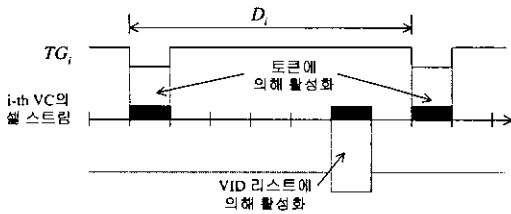


그림 3 i-th VC를 위한 주기적 토큰-기반 조절기의 타이밍 관계의 예

한편, 자신의 TG에 의해 활성화된 VCQ가 비어 있는 경우 자신에게 할당된 대역폭이 사용되지 않는 결과가 되며 이러한 일시적으로 사용되지 않는 대역폭을 *unused* 대역폭이라 한다. 한편, 각 VC들에게 할당되고 남은 대역폭은 *unallocated* 대역폭이라고 하며, 이들 *unallocated* 대역폭과 *unused* 대역폭을 합쳐서 *residual* 대역폭이라고 한다. 이러한 *residual* 대역폭은 다음 방법에 의해 사용된다. 출력 링크상의 하나의 슬롯 동안, TG들에 의해 활성화된 VCQ들과 출력버퍼가 비어있을 때(본 논문에서는 “residual 슬롯”이라고 칭한다), 만약 자신의 TG에 의해 활성화되지는 않았지만 셀이 존재하는 VCQ들이 존재하면 이들 중 하나의 VCQ 내 셀이 서비스되기 위해 선택된다. 이것은 제안된 다중화기가 work-conserving 임을 의미한다. 선택하는 방법에 있어서는 무작위(random), 가장 긴 VCQ 우선(longest VCQ first), 혹은 라운드-로빈(round-robin) 선택 등 여러 가지 방법을 택할 수 있으나 본 논문에서는 VCQ 식별자를 요소로 갖는 하나의 VID 리스트를 이용한 라운드-로빈 방법을 채택한다. 이것은 동작이 간단하고 *residual* 대역폭을 모든 VCQ들에게 공평하게 분배하기 때문이다. VID 리스트 상에 비어 있지 않은 VCQ의 VID를 관리하기 위해 각 VCQ의 플래그를 둔다. 임의의 VCQ에 하나의 셀이 도착하면 해당 플래그를 검사하고 만약 “세팅”되어 있지 않으면 자신의 VID를 리스트에 첨부하고 자신의 플래그를 “세팅”하게 된다. 따라서 플래그는 리스트 상에 해당 VID의 존재 여부를 표시하게 되며, 셀 도착 시 자신의 플래그가 “세팅”되어 있으면 VID 리스트에 자신의 VID를 첨부할 필요가 없다. Residual 슬롯에 라운드-로빈 형태로 VCQ들을 선택하기 위해 리스트의 헤드로부터 VID가 처리되고 만약 해당 VCQ에 셀이 존재하면 전송을 위해 하나의 셀이 서비스된다. 한편, 해당 VCQ가 서비스된 후 계속 셀이 존재하면 VID는 VCQ 리스트의 end-of-list에 다시 첨부되고, 만약 서비스 후 비게 되면 리스트에서 제거된다. 다음 pseudo 코드는 VID 리스트 동작을 설명한다.

**variable**

$F_i$  : integer /\* flag bit of the i-th VCQ (initially "0") \*/

**Event**

1. a cell arrivals at the i-th VCQ;
  - if ( $F_i=0$ )
    - insert i to VID list at tail
    - $F_i=1$
- endif**
2. at a residual slot;
  - while** (VID list is not null) do
    - j := remove HOL element in VID list /\* HOL : head-of-line \*/
    - if (j-th VCQ is non-empty)
      - serve the j-th VCQ
      - if (j-th VCQ is empty) /\* served cell is last one in j-th VCQ \*/
        - $F_j=0$  and delete j
      - else insert j to VID list at tail
    - endif**
    - exit while loop
  - else
    - $F_j=0$  and delete j
  - endif**
  - loop**

**3. 제안된 다중화기의 대역폭 할당 능력**

참고문헌[3]에서는 다중화기의 출력 트래픽 스트림 형태를 알아보기 위해, 두 개의 파라미터 평균률(AR : average rate)과 평균 간격(AI : average interval)을 정의하였다. AI는 트래픽 스트림의 형태를 통계적으로 충분히 파악할 수 있는 시간 간격을 의미한다. 즉 AI 시간 간격 동안 도착한 셀 수를 AI로 나누면 그 결과가 AR이 된다. 따라서, 트래픽 스트림의 형태에 따라 AI는 다음과 같은 범위를 가진다.

$$1/AR \leq AI \leq \text{전체 연결 기간}$$

만약 AI 값이 최소값  $1/AR$ 이 되면, 해당 연결은 고정율(CBR : constant bit rate) 서비스임을 나타낸다.

이상과 같은 유사한 개념으로 본 논문에서는 출력 링크 상에 각 VC를 위해 할당되는 슬롯의 형태를 표현하기 위해 두 개의 파라미터를 고려한다. 하나는 평균 서비스율 (ASR : Average Service Rate)이고 하나는 앞에서 설명한 파라미터 AI이다. 이것은 각 VC에 대한 ASR은 해당 VC에 할당된 슬롯의 전체 수를 AI로 나눔으로서 구할 수 있다. i-th VC에 대한 가능한 AI 값의 범위는

$$1/ASR \leq AI \leq (\text{i-th VC의 호 연결 기간})$$

따라서, 만약 우리가 AI 값을 최소값( $1/ASR$ )으로 유지하면 해당 VC는 고정적인 서비스율을 보장받게 된다. 서비스율  $\phi_i$  를 요구하는  $i$ -th VC의 AI 값이 적으면 적을수록 슬롯 할당 간격의 변화율이 줄어들게 된다. 따라서 AI 값은 각 VC의 성능과 출력 스트림의 형태에 영향을 미치게 되며, AI 값이 적으면 적을수록 성능과 출력 스트림에 좋은 영향을 미치게 된다[10,11].

$a_i^n$ 는  $i$ -th VC의  $n$ 번째 ( $n \geq 1$ ) 셀이 다중화기의 입력 링크 상에 도착한 도착 슬롯, 그리고  $d_i^n$ 는 이 셀이 다중화기를 빠져나가는 출력 링크 상의 출력 슬롯이라고 하자(그림 4 참조). 만약 VCQ의 입력 링크와 다중화기의 출력 슬롯이 슬롯 경계부분에서 모두 동기 되어 있다고 가정하면,  $i$ -th VC의  $n$  번째 셀의 셀 lateness [6,9],  $L_i^n$ , 는 다음과 같이 정의된다.

**Definition 1** [6,12] (Cell lateness)

$$L_i^n = d_i^n - \max(a_i^n, d_i^{n-1}) - D_i, \forall i \in R$$

그리고  $\forall n \leq 1$ , (단, 여기서  $d_i^0 = 0$  로 가정한다).

따라서 셀 lateness  $L_i^n$ 는  $i$ -th VC에 의해 발생된  $n$ -th 셀의 실질적인 서비스 시간(VCQ의 헤드와 출력 버퍼에서 대기한 시간의 합)과  $i$ -th VC를 위해 할당된 서비스 시간  $D_i$ 와 차이(단위는 슬롯)가 된다.  $L_i^n \leq 0$  ( $\forall i \in R$  그리고  $\forall n \geq 1$ ) 은 출력 VC 스트림의 버스트 성분과 셀 지연 성능에 있어 매우 바람직한 성질이다[5,6].

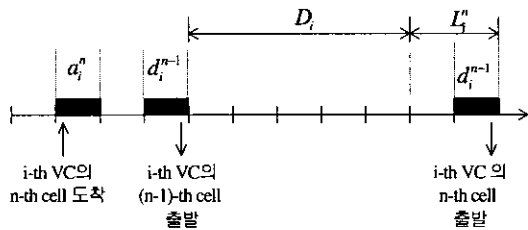


그림 4  $i$ -th VC의  $n$ -th 셀에 대한 셀 lateness 의 예

이제 각 VC에 대한 규칙 흐름제어(regular rate control)를 정의한다.

**Definition 2** 규칙 흐름제어(Regular rate control)

규칙 흐름제어 스케줄러는 per-VC에 할당된 각 서비스 시간들이 이들 중 가장 짧은 서비스 시간의  $k$  배로 동작한다. 즉  $D_i = k \times D_b, \forall i \in R$ , 여기서  $k \in \mathbb{N}$ (자연수) 그리고  $D_b = \min_{i \in R} \{ D_i \}$ .

시스템의 안정상태(stable state) ( $\sum_{i \in R} 1/D_i \leq 1$ )에서는 새로이 연결되는 VC의 TG는 이미 동작하고 있는 다른

TG로부터 발생하는 토큰들과 충돌 없이 해당 VC에 대한 토큰을 최초 발생시킬 수 있다. 따라서 규칙 흐름제어의 정의로부터 이들 TG들로부터 발생하는 토큰들은 이 후 임의의 슬롯에 동시에 발생되지 않음을 쉽게 증명할 수 있다. 따라서 규칙 흐름제어 기법에서는 임의의 슬롯에 하나의 TG 만이 자신의 VCQ를 활성화하게 된다. 이것은 출력 버퍼에 동시에 도착되는 셀이 존재하지 않음을 의미한다. 따라서 출력 버퍼에서 일시적 셀 저장이 일어나지 않기 때문에 규칙 흐름제어는  $i$ -th VC에 대해  $D_i$ 에 해당되는 고정적 서비스 시간을 보장하게 된다. 이것은 AI에 대한 최소 한계를 보장하는 것이며  $L_i^n \leq 0, \forall i \in R$  와  $\forall n \geq 1$ , 를 만족하게 된다. 즉 실질적인 서비스 시간은 할당된 서비스 시간  $D_i$ 를 결코 넘어 서지 않는다. 그러나 규칙 흐름제어는 정의로부터 알 수 있듯이 각 VC에 대한 유연한 대역폭 할당을 방해한다.

이제부터는 각 VCQ로부터 출력 버퍼에 셀이 동시에 도착하는 더욱 일반적인 경우를 고려하자. 즉 임의의 슬롯에 복수개의 토큰이 발생하게 된다. (i)  $N_j$ 는 할당 대역폭  $\phi_j$  를 요구하는 타입  $j$  VC의 개수( $j=1,2, \dots, m$ )라고 표시하고  $r = \sum_{j=1}^m N_j$  을 만족한다, (ii)  $TG_i, i \in R$ , 는  $n \times D_i + s_i, n=0,1, \dots$ 에서 토큰을 발생시키며,  $s_i$  는  $TG_i$  의 최초 토큰 발생 슬롯에 해당한다. 그리고 (iii)  $TG_i$  은 마지막으로 활성화된 TG이다. 한편,  $M^* = lcm_{i \in R} (D_i)$ , 여기서 lcm은 최소공배수(least common multiplier)다.

셀 스케줄링 기법은 per-VC에 대한 셀 손실을, 셀 지연, 그리고 셀 지연 변이에 대해 양 끝점(end-to-end) 간 정해진 한계를 보장할 수 있어야 한다. 비록, work-conserving 성질은 평균 셀 지연과 평균 버퍼 점유율은 감소시키지만 per-VC 셀 지연의 상위 한계(upper bound)를 제공하는 것은 아니다. work-conserving 성질을 가진 제안된 RCC 다중화기는 이러한 일반적인 경우에 대해서도 per-VC 셀 지연을 제한하게 된다. 이것은 다른 VC 트래픽 특성에 상관없이 정해진 셀 수가 지체 없이 전송되도록 보장된다는 것을 보임으로 가능하다. 이것은 결국 하나의 서비스가 다른 서비스들로부터 분리될 수 있다는 것을 보인다. 이와 같은 중요한 성질은 각 VC의 셀 lateness가 제한(bound)된다는 것을 보임으로서 가능하다. 특히, 앞에서 언급한 바와 같이 정해진 최소 전송 셀 수의 보장을 위한 간격(AI)이 적으면 적을수록 성능이 좋아지게 된다[10,11].

이제 제안된 RCC 다중화기가  $i$ -th VC ( $\forall i \in R$ )에 대해  $AI = M^*$  기간에 해당  $ASR (= \phi_i)$ 을 보장하고 다

른 VC 트래픽 특성에 상관없이 각 VC의 셀 lateness가 제한됨을 보인다.

**Lemma 1** 제안된 RCC 다중화기는  $i$ -th ( $\forall i \in R$ ) VC에 대해 기간  $AI = M^*$ 에 해당  $ASR (= \phi_i)$ 을 보장한다.

(증명) 최악의 상태(worst-case scenario)를 고려하자. 이것은 모든 VCQ들이 계속 비어 있지 않은(backlogged) 상태에 있을 때 발생한다. 이와 같은 경우 제안된 RCC 다중화기(그림 2 참조)는 그림 5에 해당한다. 결국 그림 5에서, VCQ 읽기 버스 상의 하나의 토큰은 다중화기 출력 링크 상의 각 슬롯에서 해당 VCQ를 서비스하게 된다. 즉  $i$ -th VC를 위한 VCQ 읽기 버스 상의 토큰들은 출력 링크 상에 할당된 슬롯에 해당된다. 토큰 버퍼는 이산시간  $N_1 \times D_1 + \dots + N_m \times D_m / D_1$  큐잉 모델로 동작한다.

토큰버퍼에서  $TG_j$ 의  $n$ -th 토큰의 대기시간,  $W_j^n$ , 은 다음과 같이 구할 수 있다.

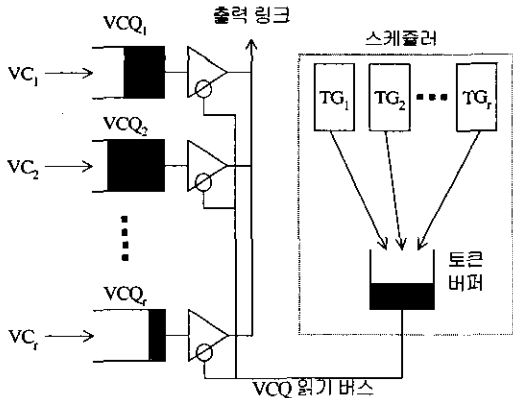


그림 5 제안된 RCC 다중화기의 최악의 상태(worst-case)에서의 동일 모델(equivalent model)

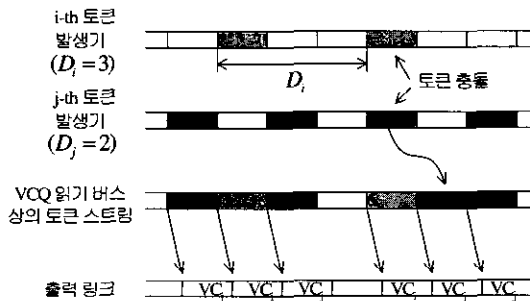


그림 6 타이밍 관련 예

$$W_j^n = X_{nD_j + s_j - 1} + \sum_{i=1}^{j-1} I_i^{nD_j + s_j} \quad (1)$$

여기서,  $X_t$ 는  $t$ -th 슬롯의 끝에서 토큰 버퍼 내 토큰 수를 나타낸다. 그리고  $I_i^t$ 는 슬롯  $t$ 에서  $TG_i$ 의 토큰 발생에 대한 indicator 랜덤 변수이다. 즉

$$I_i^t = \begin{cases} 1 & \text{if } \text{mod}(t - s_i, D_i) = 0 \\ 0 & \text{if } \text{mod}(t - s_i, D_i) \neq 0 \end{cases} \quad (2)$$

여기서  $\text{mod}(a,b)$ 는 분수  $a/b$ 의 나머지에 해당한다. 상기 식 1의 오른쪽 두 번째 항은 동일 슬롯에 발생된 토큰들은  $TG_i, i=1, \dots, j$ 의 아래 침차  $i$ 의 '올림차순'으로 토큰버퍼에 저장된다고 가정한 것이다. 즉  $i$  값이 적은  $TG$ 로부터 발생된 토큰이 토큰 버퍼에 먼저 저장된다.  $iTI_j^n$ 는 VCQ 읽기버스 상에  $TG_j$ 로부터 발생된  $n$ -th와  $(n+1)$ -th 토큰 사이 간격을 나타낸다. 이것은 출력 링크 상에  $j$ -th VCQ로부터  $n$ -th와  $(n+1)$ -th 전송 셀 사이 시간 간격에 해당된다. 한편,  $iTI_j^n$ 은 다음과 같이 구해진다.

$$iTI_j^n = D_j + W_j^{n+1} - W_j^n \quad (3)$$

이제 모든  $TG$ 들이 활성화되어 있는 경우, VCQ 읽기 버스 상에 나타나는  $TG_j$ 의  $n$ -th 토큰( $n \times D_j + s_j \geq s_j$ 과  $j \in R$ 을 만족하는 모든  $n$ 과  $j$ 에 대해)으로부터 연속적인  $m_j (= M^* / D_j)$  개 토큰을 고려해 보자.  $j$ -th VCQ를 위한  $m_j$  개의 연속적인 토큰에 대한 VCQ 읽기 버스 상의 시간간격은 다음과 같이 계산된다.

$$iTI_j^n + iTI_j^{n+1} + \dots + iTI_j^{n+m_j-1} = m_j D_j + W_j^{n+m_j} - W_j^n = M^* \quad (4)$$

식 4의 두 번째 등식은  $I_j^t$ 와  $X_t$ 의 주기가  $M^*$  ( $\forall j \in R$  그리고  $\forall t \geq s_j$ )인 주기성(부록의 증명 과정 참조)으로부터 확인할 수 있다. 따라서

$$W_j^n = W_j^{n+m_j} \quad (5)$$

궁극적으로,  $j$ -th VC,  $j \in R$ ,에 대해 해당 VCQ는  $M^*$  동안에  $m_j$  번 서비스 받게된다. 따라서 상기 레마는 성립한다. □

**Lemma 2** 제안된 RCC 다중화기는 각 VC에 대한 셀 lateness가 제한된다.

(증명) 레마 1에서와 같이 최악의 상태를 가정하고  $j$ -th VC의  $k$ -th 셀이  $TG_j$ 의  $n$ -th 토큰에 의해 스케줄된다고 가정하자. 한편, 정의 1로부터  $L_j^k$ 는  $L_j^k \leq (iTI_j^{k-1} - D_j) = (W_j^k - W_j^{k-1})$ 의 관계식에 의해 제한된다. 따라서  $j$ -th VC의 최대 셀 lateness는 다음과 같이 주어진다.

$$L_j \leq \max_{k \leq n(k + \frac{M^*}{D_j})} (W_j^n - W_j^{k-1}), \forall j \in R, \forall k \geq s_i \quad (6)$$

여기서,  $k \geq s_i$  는 모든 TG들이 모두 활성화되어 있음을 보여준다. 앞에서와 같이토른 버퍼에서의 토른 대기시간들이 주기  $M^*$  를 가진 주기성을 갖기 때문에 (부록의 증명 과정 참조)  $M^*$  슬롯의 토른 발생 시간만 고려하여도 된다. 따라서 상기 레마는 성립한다. □

본 논문에서 제안된 다중화기는 하나의 카운터에 의한 주기적인 토른 발생기와 residual 대역폭을 위한 라운드-로빈 선택에 의존하기 때문에 하드웨어 구성 면에서 기존의 셀 스케줄링 기법들[2-9]이 가지고 있는 하드웨어 구성의 어려운 점 (타임 stamping, deadline 비교, 정렬 큐, 그리고 프레임 관리 등)을 극복할 수 있다.

#### 4. 시뮬레이션

제안된 RCC 다중화기의 성능을 서비스 분리 능력 및 출력 스트림의 버스티성 감소의 관점에서 보여주기 위해, 하나의 ATM 노드에 다양한 트래픽을 대상으로 수행한 시뮬레이션 결과를 제시하고, 이들 결과를 참고문헌[6]에서 제시된 HOL-EDD가 적용된 다중화기의 결과와 비교한다. 참고문헌[6]에서는 HOL-EDD 기법을 기존의 다양한 셀 스케줄링 기법과 비교한 자료가 제시되고 있다. 따라서 이를 통해 본 논문에서 제시된 RCC 다중화기의 성능을 기존 셀 스케줄링 기법과 비교해 볼 수 있다.

##### 4.1 Per-VC 버스티성 감소

통계적 다중화기의 셀 스케줄링 기법은 출력 스트림의 특성을 변화시킨다. 먼저, 그림 7에 보여지는 시뮬레이션 환경에서 참고문헌[6]에서와 같이 출력 스트림의 버스티성의 합을 조사한다. 여기서 사용되는 트래픽 특성은 비교의 편의성을 위해 참고문헌[6]에서 사용된 것과 동일하게 구성하였다(표 1 참조). 스트림의 버스티성은 해당 스트림에 할당된 서비스율과 동일한 서비스율을 가진 테스트 큐의 최대 backlog로 추정된다(그림 7 참조).

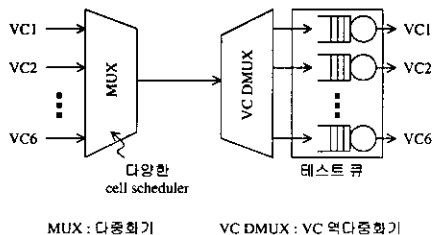


그림 7 출력 스트림의 버스티성 측정을 위한 시뮬레이션 환경

표 1 시뮬레이션-1의 트래픽 호원

VC	$D_i$	$B$	$I_c$	$I_b$
1	6	U(1,10)	D(5)	D(20)
2	5	U(1,10)	D(5)	D(20)
3	7	U(1,10)	D(5)	D(20)
4	6	U(1,10)	U(1,9)	U(1,39)
5	6	U(1,10)	U(1,9)	U(1,39)
6	7	U(1,10)	U(1,9)	U(1,39)

ON/OFF 호원은 트래픽 파라미터 ( $B, I_c, I_b$ )로 표현된다. 여기서,  $B$ 는 ON 버스트 구간동안 발생되는 셀 수를,  $I_c$ 는 ON 버스트 구간동안의 셀 사이 간격(intecell interval), 그리고  $I_b$ 는 OFF 구간의 길이를 나타낸다. 표 1에서  $D_i$ 는 제2절에서 정의하였으며  $U(a, b)$ 는  $a$ 와  $b$  사이 ( $a, b$ 를 포함) uniform 랜덤 변수를 나타내고  $D(a)$ 는 값이  $a$ 인 deterministic 정수 값을 표현한다.

시뮬레이션은  $10^8$  슬롯 동안 수행되었으며 표 2는 테스트 큐들의 최대 backlogs와 각 VC의 최대 셀 lateness를 보여준다. 이러한 시뮬레이션 결과들로부터 우리는 본 논문에서 제안된 RCC 다중화기의 성능이 HOL-EDD 기법에 매우 근접해 있음을 확인할 수 있다. 참고문헌[6]에서는 HOL-EDD 기법이 Virtual Clock(VCLK), Packetized Generalized Processor Sharing(PGPS) 등과 같은 기존의 다른 셀 스케줄링 기법과 비교해 최상의 per-VC 버스티성 감소 특성을 가지고 있음을 보인다. 따라서 제안된 RCC 다중화기의 상대적인 비교가 가능하다.

표 2 각 VC의 출력 스트림의 버스티성과 최대 셀 lateness

VC	출력 VC 스트림의 버스티성		최대 cell lateness	
	RCC 다중화기	HOL-EDD	RCC 다중화기	HOL-EDD
1	4	3	1	0
2	2	2	1	1
3	4	4	1	1
4	33	32	1	1
5	34	33	2	1
6	60	60	1	1
전체	137	134	7	5

반면에, 하드웨어 실현의 관점에서는 RCC 다중화기는 HOL-EDD 기법에 비교해 대규모 혹은 고속의

ATM 노드를 구성할 때 유리하다. 이것은 RCC 다중화기가 TG들을 위한 카운터들과 VID 리스트를 위한 큐를 필요로 하는 반면, HOL-EDD 기법은 수 마이크로 초(microsecond) 내에 현재 연결되어있는 모든 연결들의 deadline들을 비교해야 한다. 이러한 deadline 비교는 결국 연결수가 증가하거나 혹은 링크 속도가 증가하게 되면 하드웨어 실현을 극도로 어렵게 하는 요인으로 작용한다.

4.2 서비스 분리

트래픽 분리 특성을 확인하기 위해, 본 시뮬레이션에서는 세 개의 목적(target) 호원(VC1, 2, 그리고 3)과 cross 트래픽(VC 4)의 효과를 보이기 위해 하나의 베르누이(bernoulli) 호원을 사용한다. 각 호원의 트래픽 파라미터는 표 3과 같다.

표 3 시뮬레이션을 위한 트래픽 호원 - II

번호	트래픽 타입	트래픽 파라미터
VC 1	CBR	$I_c=5$
VC 2	VBR(On/Off)	$B \sim U(1, 19), I_c=3, I_b=20$
VC 3	VBR(On/Off)	$B \sim U(1, 39), I_c=3, I_b=40$
VC 4	Bernoulli process	$I_c \sim \text{Geo}(0.2) - \text{Geo}(0.5)$

표 3에서 VC1은 고정 비트율(CBR), VC2와 VC3은 변화 비트율 (VBR) 호 원으로서 평균 입력 부하가 0.2이고, VC4는 베르누이 프로세스로 평균 입력 부하가 0.2에서 0.5 사이에서 변화한다. Cross 트래픽의 평균 입력 부하가 0.4를 넘어서게 되면 전체 입력 부하가 출력 링크 능력을 초과함에 따라 어떤 큐들의 길이는 무한히 증가될 것이 예상된다. 본 시뮬레이션에서는 하나의 ATM 노드는 유한한 크기의 버퍼를 가진다고 가정하고 이와 같은 과부하 상태에서는 하나의 FIFO 다중화기를 가정하면 모든 VC로부터 입력되는 셀들이 폐기된다. 그러나 제안된 RCC 다중화기는 per-VC 대역폭 보장에 따라 VC4로부터 입력되는 셀 만이 폐기되게 된다.

표 4는 유한한 크기의 큐(800 셀)를 가진 하나의 ATM 노드에서 표 3으로부터 각 호 원에 대한 평균/최대 셀 지연을 보여준다. 제안된 RCC 다중화기 경우, 각 VC는 동일한 크기의 버퍼(200 셀)와 대역폭 보장 ( $\phi_i=0.2, 1 \leq i \leq 4$ )을 가지게 되고 할당되지 않은 대역폭 0.2는 모든 VC들에 의해 공유된다. Cross 트래픽 VC4의 강도가 0.2에서 0.3으로 증가함에 따라 residual 대역폭의 감소가 발생하고 제안된 다중화기의 목적 호

원들의 지연 성능이 악화되기 시작한다. 그러나 FIFO 다중화기와는 달리, 과부하 상태(cross 트래픽의 강도가 0.4를 초과)에서도 제안된 다중화기는 목적 호 원들의 셀 지연 성능을 심각하게 감소시키지는 않는다. 즉 목적 호 원들에 대해 최대 셀 지연을 제한하게 된다. 따라서 제안된 다중화기는 각 VC에 대해 명확히 대역폭을 보장하게 되고 우리가 원하는 트래픽 분리를 가능케 한다. 표 4에서 보여지는 바와 같이 HOL-EDD 기법 역시 제안된 RCC 다중화기와 같이 동작하지만, 앞에서 언급한 바와 같이 모든 연결들에 대해 매 수 마이크로 초마다 deadline들을 비교해야 하는 문제점이 있다.

표 4 평균 및 최대 셀 지연 특성

다중화기 타입	target 스트림	항목	cross 트래픽 부하 (VC 4)			
			0.2	0.3	0.4	0.5
FIFO	VC 1	AVG	2.21	4.23	750.24	797.54
		MAX	21	43	800	800
	VC 2	AVG	2.44	4.51	750.34	797.67
		MAX	20	44	800	800
	VC 3	AVG	2.63	4.92	750.63	798.08
		MAX	20	44	800	800
	VC 4	AVG	2.46	4.51	750.15	797.37
		MAX	20	44	800	800
HOL-EDD	VC 1	AVG	1.09	1.06	1.03	1.03
		MAX	4	4	4	3
	VC 2	AVG	2.68	3.63	3.97	3.98
		MAX	18	18	17	17
	VC 3	AVG	3.24	5.22	6.13	6.13
		MAX	29	29	30	30
	VC 4	AVG	2.71	7.02	395.98	504.01
		MAX	47	93	623	638
RCC 다중화기	VC 1	AVG	1.36	1.45	1.51	1.51
		MAX	2	2	2	2
	VC 2	AVG	2.63	4.08	5.50	5.50
		MAX	19	19	19	19
	VC 3	AVG	2.91	5.08	7.27	7.28
		MAX	27	30	32	32
	VC 4	AVG	2.84	6.56	393.93	501.92
		MAX	48	90	622	636

5. 실현의 관점

앞에서 언급한 바와 같이 셀 스케줄링을 위해 매 타임 슬롯마다 각 VC들의 deadline(혹은 타임 스템핑) 비교를

하드웨어로 실현하는 것은 현실적으로 ATM 노드에서는 결코 쉬운 작업이 아니다. 참고문헌[9]에서는 이러한 문제점을 해결하기 위해, 병렬화를 이용한 connectionlist HOL-EDD 서비스 스케줄러를 제안하였다. 제안된 스케줄러는  $n$ -비교 소자( $n$ -comparison element)를  $n$ -ary 트리 상호 연결을 이용하여  $O(1)$  계산 복잡도를 가진 동작 특성을 보인다. 그러나 시간적으로 변화하는 연결 VC의 수에 따라 다이내믹(dynamic)하게 트리 연결 구성을 변화시킨다는 것은 거의 불가능함으로 고정된 깊은 depth 트리 상호 연결이 ATM 노드의 각 출력 포트마다 요구된다. 이것은 트리 프로세싱에서 추가적인 셀 지연을 발생시키며, 이를 실현하기 위해서는 대단히 큰 하드웨어 구성이 요구된다.

본 논문에서 제안된 RCC 다중화기의 계산 복잡도는  $O(r)$ 이다. 여기서  $r$ 은 현재 출력 링크 상에 연결된 VC의 수다. 이것은 residual 슬롯 동안에 VID 리스트로부터 셀이 존재하는 VCQ를 선택하기 위한 최악의 복잡도(worst-case complexity)가  $O(r)$  이기 때문이다. 그러나 제2장에서 설명한 바와 같이 이들 비교는 VCQ 내 셀의 존재 여부만 판단(버퍼 empty 플래그 비교)하기 때문에 비교기 구현이 간단하며 따라서 하나의 슬롯 동안 비어 있지 않은 VCQ를 찾을 때까지 VID 리스트를 여러번 액세스가 가능하다. 뿐만 아니라 하나의 residual 슬롯 동안 비어 있는 VCQ의 VID를 재거함으로서 VID 리스트는 비어 있지 않은 VCQ의 VID를 유지하는 경향이 아주 강하다(2장 참조). 따라서 VID 리스트는 거의 비어 있지 않은 VCQ의 VID를 가지게 되며, 제안된 RCC 다중화기는 실 시간성을 유지하게 된다.

## 6. 결론

본 논문은 명확하게 per-VC 대역폭 할당이 가능한 ATM 망에서 흐름제어 셀 다중화기를 제안하였다. 제안된 다중화기는 work-conserving 특성을 가짐으로서 망 자원을 효과적으로 사용하게 된다. 특히, 제안된 다중화기는 카운터에 의한 "주기적인 토큰 발생기"와 residual 대역폭을 위한 "라운드-로빈"으로 동작함으로써 기존의 셀 스케줄링 기법들이 이용하고 있는 timestamping, deadline 비교, 정렬 큐, 그리고 프레임 관리에 의한 하드웨어 실현의 어려움을 극복할 수 있다. 또한 제안된 다중화기의 주요 특성을 보이기 위해 보장된 서비스율의 평균 구간과 셀 lateness를 이용한 실질적인 서비스 시간을 논하였다. 시뮬레이션을 이용한 최대 셀 lateness와 셀 지연과 같은 성능 비교 측정을 통해 제안된 다중화기가 명확하게 할당된 대역폭에 따라 각 VC를 효과적

으로 분리하고 있으며 오동작(misbehaving) 호 원들의 충격으로부터 보호할 수 있음을 보였다.

## 참고 문헌

- [1] Kang, K. and Kim, C., "Performance Analysis of Statistical Multiplexing of Heterogeneous Discrete-time Markovian Arrival Processes in an ATM Network," Computer Communications, Vol.20, No. 11, pp.970-978, 1997.
- [2] Bolla, R. Davoli, F. and Marchese, M., "Bandwidth Allocation and Admission Control in ATM Networks with Service Separation," IEEE Communications Magazine, pp.130-137, 1997
- [3] Zhang, L., "VirtualClock: A new traffic control algorithm for packet switched networks," ACM Trans. on Computer Systems, Vol.9, No.2, pp.101-124, 1991
- [4] Parekh, A.K. and Gallager, R.G., "A Generalized Processor Sharing Approach to Flow Control in Integrated Services Networks: The Single-Node Case," IEEE/ACM Trans. on Networking, Vol.1, No.3, pp.344-357, 1993
- [5] Zhang, H. and Ferrari, D., "Rate-Controlled Static-Priority Queueing," Infocom'93, 1993
- [6] Vishnu, M. and Mark, J.W., "HOL-EDD: A Flexible Service Scheduling Scheme for ATM Networks," Infocom'96, 5d.3.1-5d.3.8, 1996
- [7] Kalmanek, C.R., Kanakia, H. and Keshav, S., "Rate Controlled Servers for Very High-Speed Network," Infocom'90, pp.300.3.1-300.3.9, 1990
- [8] Golestani, S.J., "A Framing Strategy for Congestion Management," IEEE J. Select Areas Commun., Vol.9, No.7, pp.1064-1077, 1991
- [9] Vishnu, M and Mark, J.W., "A connection list implementation of the VCQ ATM switch," ICC'95, pp.1115-1121, 1995
- [10] Kang, K., Steyaert, B. and Kim, C., "A Simple Relation between Loss Performance and Buffer Contents in a Statistical Multiplexer with Periodic Vacations," IEICE Trans. Communications, Vol. E80-B, No.11, pp.1749-1752, 1997
- [11] Kang, K. and Steyaert, B., "Bound Analysis for WRR Scheduling in a Statistical Multiplexer with Bursty Sources," Telecommunication System 12, pp.123-147, 1999

## 부록

본 부록은 레마1과 레마2를 증명하기 위해 필요한 그림 5의 토큰 버퍼의 큐잉 모델에서 토큰 버퍼 내 토큰 수와 토큰의 대기 시간이 주기성을 갖는 것을 증명한다.



이 모델은 제 3장에서 설명한 바와 같이 이산시간의  $N_1 \times D_1 + \dots + N_m \times D_m / D_1$  큐잉 시스템으로 모델링 된다. 여기에서 모든 TG들은 항상 활성화되어 있다고 가정한다. t-슬롯 끝 지점에서 토큰 버퍼 내에 있는 토큰의 수  $X_t$ 를 결정하고  $X_t$ 의 행위를 조사한다.

$A_t$ 는 슬롯 t에 도착하는 전체 토큰의 수를 나타낸다면 다음과 같다.

$$A_t = \sum_{i=1}^m I_i^t \quad (7)$$

여기서,  $I_i^t$ 를 슬롯 t에서 TG<sub>i</sub>로부터 토큰 발생에 대한 indicator 랜덤 변수다 (식 2 참조). 모든 TG들은 자신의 주기적 토큰 발생 주기를 기준으로 토큰을 발생시키기 때문에  $M^* (= lcm_{i \in R} D_i)$  슬롯 구간 동안에 관찰되는  $A_t$ 는 주기적이다 즉,

$$A_t = A_{t+M^*} \quad (8)$$

따라서  $X_t$ 는 다음과 같이 재귀적(recursively)으로 표현된다.

$$X_t = [X_{t-1} + A_t - 1]^+, t > 0, \quad (9)$$

여기서,  $[x]^+ = (0, x)$  이고  $X_0 = 0$ 이다. 반복적인 대입에 의해 상기 식은 다음과 같이 표현된다.

$$\begin{aligned} X_t &= \max(A_t, A_t + X_{t-1} - 1) \\ &= \max(A_t, A_t + A_{t-1} - 1, A_t + A_{t-1} + X_{t-2} - 2) \quad (10) \\ &= \max_{0 \leq s \leq t} \left( \sum_{j=t-s}^t A_j - s \right) \end{aligned}$$

Excess 함수  $\phi_t(s)$ 를 정의하면

$$\phi_t(s) = \sum_{j=t-s}^t A_j - s, 0 \leq s \leq t \quad (11)$$

식 10으로부터  $X_t$ 는 다음과 같이  $\phi_t(s)$ 의 항으로 표현될 수 있다.

$$X_t = \max_{0 \leq s \leq t} \phi_t(s) \quad (12)$$

$\phi_t(0) = A_t$  임으로 이것은 다음과 같다.

$$\begin{aligned} \phi_t(M^*) &= \sum_{j=t-M^*}^t A_j - M^* = \sum_{j=t-(M^*-1)}^t A_j - M^* \\ &= A_{t-M^*} + \sum_{j=1}^{M^*-1} \frac{D_j}{M^*} - M^* \quad (13) \\ &= M^* \left( \sum_{j=1}^{M^*-1} \frac{D_j}{M^*} - 1 \right) + A_t \\ &\leq A_t = \phi_t(0), \end{aligned}$$

여기서, 네 번째 등식은  $\sum_{j=1}^{M^*-1} 1/D_j \leq 1$ 의 조건으로부터 만족된다. 한편, 상기 식과 같은 동일한 전개들 통하여  $0 \leq s \leq M^*$ 와  $n > 0$ 에 대해  $\phi_t(s+nM^*) \leq \phi_t(s)$ 가 만족된다. 따라서 식 12는  $0 \leq s \leq M^*$ 에서  $\phi_t(s)$ 의 값으로 결정된다. 따라서 식 12는 다음과 같이 쓰여질 수 있다.

$$X_t = \max_{0 \leq s \leq M^*} \phi_t(s) \quad (14)$$

따라서, 식 8, 식 11, 그리고 식 14로부터  $X_t$ 는 주기기  $M^*$ 인 주기성 ( $X_t = X_{t+M^*}$ )을 갖게 된다.



강 구 홍

1985년 경북대학교 전자공학과 졸업(공학사). 1985년 ~ 1993년 한국전자통신연구소 선임연구원. 1990년 충남대학교 전자공학과 졸업(공학석사). 1998년 포항공과대학교 전자계산학과 졸업(공학박사). 1998년 ~ 1999년 한국전자통신연구원 선임연구원. 2000년 ~ 현재 서원대학교 컴퓨터정보통신공학부 정보통신공학과 전임강사. 관심분야는 트래픽 제어, 성능평가, 컴퓨터 네트워크



박 상 조

1983년 서울대학교 제어계측공학과 졸업(공학사). 1985년 서울대학교 제어계측공학과 졸업(공학석사). 1984년 ~ 1991 삼성전자 종합연구소 과장. 1992년 ~ 2000년 한국전자통신연구원 선임연구원. 1995년 ~ 1996년 일본 국립 오사카대학교 연구생. 1999년 일본 국립 오사카대학 졸업(공학박사). 2000년 ~ 현재 서원대학교 컴퓨터정보통신공학부 정보통신공학과 전임강사. 관심분야는 광무선 액세스 네트워크, 광 CDMA 방식, ATM 네트워크