

## ATM망에서 주기적 대역폭 할당을 이용한 최선형 트래픽의 전송을 제어 기법

正會員 權泰境\*, 崔陽熙\*

### Rate Control Scheme with Periodic Bandwidth Allocation(RCPBA) for the Best-effort Traffic in ATM Network

Kwon Tae Kyoung\*, Choi Yang Hee\* Regular Members

#### 要 約

본 논문에서는 ATM망에서 주기적으로 대역폭을 할당하여 종점간에 전송율을 조정하는 새로운 흐름 제어(flow control)기법을 제안한다. 제안된 기법에서는 수신자가 주기적으로 대역폭 할당 셀(Rate Allocation Cell)을 송신자에게 보내면 네트워크의 중간 노드들은 이 셀에 허용가능한 대역폭 할당을 표시하고 송신자는 할당된 대역폭에 맞추어 전송율을 조정한다. 제안된 기법의 성능은 시뮬레이션을 통해 기존의 흐름 제어 기법들과 비교하고 분석하였다. 시뮬레이션 환경은 주로 보장형 트래픽(guaranteed traffic)을 서비스하고 남은 잔여 대역폭을 얼마나 잘 활용하여 최선형 트래픽(best-effort traffic)을 지원할 수 있는가를 파악하는 데에 초점을 두어 설정하였다. 시뮬레이션 결과 보장형 트래픽의 도착 시간이 포아송(Poisson) 분포일 때, 제안된 기법이 기존의 기법보다 효율적인 전송을 하는 것으로 분석되었다.

#### ABSTRACT

In this paper we propose a new flow control scheme for ATM(Asynchronous Transfer Mode) network, called RCPBA(Rate Control with Periodic Bandwidth Allocation) in which the end-to-end data transmission rate is periodically adjusted in accordance with the congestion in the transit nodes. In RCPBA, the receiver periodically sends back to the sender the RAC(rate allocation cell), where the transit cells write the maximum permissible bandwidth for the connection, and the sender adjusts the transmission rate accordingly for the next period. The proposed scheme is compared with the existing flow control schemes through simulation. The simulation environment is designed to highlight the efficient use of the bandwidth, remaining after servicing the guaranteed traffic, for the best-effort traffic. Simulation results show that the proposed scheme performs better when the guaranteed traffic is of Poisson type.

\* 서울대학교 컴퓨터공학과  
論文番號 : 95089-0227  
接受日字 : 1995年 2月 27日

## 1. 서론

최근에 멀티미디어 데이터와 관련된 다양한 서비스를 고속의 통신망을 통해서 사용자에게 제공하려는 연구<sup>(1)</sup>가 진행되고 있다. 그 중에서 광대역통신망(B-ISDN : Broadband Integrated Services Digital Network)에 관한 연구는 국제전기통신연합(ITU : International Telecommunication Union)과 ATM Forum 등 표준화 위원회를 통해서 광대역망의 근간으로 채택된 ATM망에 대한 표준을 정하기 위한 연구<sup>(2, 3, 4, 5, 6, 7, 8)</sup>가 활발히 진행되고 있다.

하나의 통신망에서 여러 서비스를 사용자에게 제공하기 위해서는 우선 네트워크가 다양한 서비스를 제공하면서도 각 서비스의 품질을 보장해 주어야 하는데, 이를 위해서는 네트워크의 트래픽이 혼잡(congestion)을 일으키지 않도록 제어하는 것이 필요하다. 여기서 혼잡이란 단순한 버퍼 오버플로우에 의한 셀 손실이 아니라 네트워크가 협상된 서비스 품질을 만족시키지 못하는 상태를 말한다<sup>(3)</sup>. 혼잡은 대부분 네트워크 트래픽의 변동 때문에 일어나는데, 이러한 네트워크 트래픽의 유동적인 흐름을 제어해야 각각의 서비스 품질을 만족시킬 수 있기 때문에 이를 위해 혼잡 상황을 방지하는 데에 중점을 두는 트래픽 제어, 혼잡이 발생했을 때 신속하게 대처와 혼잡의 피해를 최소화하는 것에 목적을 둔 혼잡 제어, 그리고 실제 데이터 트래픽을 어떻게 전송할 것인가에 대한 흐름 제어 기법들에 대해 초점이 모아지고 있다<sup>(12, 3, 4, 5, 6, 7, 8)</sup>.

ATM망에서의 혼잡을 일으키지 않게 하기 위해서는 통신망에서의 트래픽에 대해 흐름 제어를 하는 것이 필요한데, 이는 트래픽이 어떤 서비스에 속한 것이냐에 따라 차이가 있다. 주문형 비디오(VOD : Video On Demand)나 화상 회의(Video-conference)와 같은 보장형 서비스는 실시간 멀티미디어 트래픽을 다루며, 따라서 네트워크는 일정한 서비스 품질(QoS : Quality of Service)을 항상 제공할 수 있어야 한다. 여기서 서비스 품질은 대역폭, 지연시간, 지터(jitter) 등의 인자로 정의된다<sup>(3)</sup>. 실시간 멀티미디어와 같은 보장형 서비스를 위하여는 각 노드에서 링크의 대역폭이나 버퍼 등의 망 자원을 예약 및 관리(NRM : Network Resource Management)하여 종점간 지연 시간(end-to-end delay)이나 에리움 등의 서비스 품질을

보장하는 것이 필요하다. 이를 위해 호 수락 제어(CAC : Call Admission Control)시 각 가상 채널의 트래픽은 자신의 트래픽 특성을 나타내는 파라미터(평균 대역폭, 피크 전송율 등)를 제시하고 네트워크는 트래픽 전송시 그 파라미터 값에 맞게 전송되는지 검사하고 또 파라미터대로 전송되도록 사용 인자 제어(UPC : Usage Parameter Control)를 하는 것이 필요하다.

반면에 화일 전송, 원격 명령 등과 같은 최선형 서비스(best-effort service)의 경우는 미리 네트워크의 자원을 예약하는 것보다는 트래픽이 생성될 때마다 네트워크의 상황에 맞추어 혼잡을 일으키지 않으면서 네트워크의 자원(대역폭, 버퍼 등)을 효율적으로 이용하여 데이터를 전송하는 것이 중요하다. 따라서 일정한 서비스 품질을 보장해 주어야 하는 보장형 서비스의 트래픽보다는 네트워크의 상황에 따라 가능한 최선으로 서비스를 해 주면 되는 최선형 서비스의 트래픽에 대해 어떻게 네트워크에서 혼잡을 일으키지 않으면서 흐름 제어를 할 것인가에 대해 많은 연구가 있었는데, 그 중 대표적인 것은 Kung의 N23기법<sup>(5, 6)</sup>과 Hluchyj의 CLRC(Closed Loop Rate Control)기법<sup>(7, 8)</sup>으로서 세부적인 메카니즘에 대해서는 2장에서 상세히 설명하였다.

흐름 제어 기법에서는 중간 노드에서 어떻게 혼잡 상황을 파악하고 또한 얼마나 신속하게 송신자와 수신자에게 대처할 것인가가 가장 핵심이 되는 문제이다. 또한 여러 기법들이 송수신자와 중간 노드들에게 얼마나 처리 부하를 줄 것인가가 구현 및 성능 면에서 초점이 되고 있다. 따라서 효율적인 흐름 제어 기법이 되기 위해서는 네트워크의 노드들에게 오버헤드를 주지 않으면서도 중간 노드들의 대역폭 이용율이나 파워(처리율/지연시간) 등의 성능을 높이는 것이 중요하다. N23기법은 대역폭 이용율 등의 성능은 좋으나 각 가상 채널에 대해 처리 부하가 너무 많고 또한 트래픽이 버스트하게 전송이 되는 단점, 즉 윈도우 기법으로서의 단점을 가진다. CLRC기법은 전체적인 대역폭 이용율이 너무 낮고 비교적 짧은 기간의 평균 큐 길이에 의존하여 혼잡여부를 감지하므로 실제 잔여 대역폭을 잘 파악하지 못하며 조정되는 전송율의 양이 네트워크의 상태 변화에 비해 작은 단점이 있다.

따라서 본 논문에서는 비교적 처리 부하가 적으면서도 중간 노드들의 잔여 대역폭을 비교적 정확히 파악하여

송신자의 전송율을 변화시키는 주기적 대역폭 할당을 이용한 전송을 제어 기법(RCPBA : Rate Control with Periodic Bandwidth Allocation)을 제안하고 다양한 시뮬레이션 환경에서 각각의 흐름 제어 기법이 어떠한 성능을 나타내는지 비교 분석해 보고자 한다.

2장은 기존의 흐름 제어 기법인 Kung의 N23기법<sup>(5,6)</sup>과 Hluchyj의 CLRC기법<sup>(7,8)</sup>에 대해 각각 2.1절, 2.2절에서 설명하고, 3 장은 제안된 주기적 대역폭 할당을 이용한 전송을 제어 기법(RCPBA: Rate Control with Periodic Bandwidth Allocation)에 관한 내용으로, 3.1절에서는 동작원리를, 3.2 절에서는 장 단점 분석을, 3.3절에서는 오버플로우를 방지하기 위한 버퍼 크기를 분석하였다. 4장은 시뮬레이션 환경을 정의하고 있다. 4.1절은 시뮬레이션의 환경, 즉 네트워크의 위상(topology) 구성과 보장형 트래픽의 부하 설정을 다루고, 4.2절은 시뮬레이션의 성능 평가 기준에 대해 설명하며, 4.3, 4.4, 4.5절은 각각 N23기법, CLRC 기법, RCPBA기법에서 성능에 영향을 끼치는 제어 인자에 대해 설명한다. 5장은 시뮬레이션에 따른 결과 및 분석을 제시하고 있다. 6장에서는 결론을 담았다.

## II. 관련 연구

### 2.1 N23(5,6)

하버드대학의 Kung이 제안한 N23기법은 각 가상 채널(VC : Virtual Channel)에 대해 크레딧 기반의 링크간 흐름 제어를 한다. 즉 가상 채널의 전체 연결을 이루는 각 링크 단위로 송신자와 수신자를 설정하고 수신자가 매 N2개의 데이터 셀을 전송한 후 현재 버퍼의 남은 크기인 크레딧(그림 1의 C)를 송신자에게 전송하면 송신자는 받은 크레딧에서 그 직전 왕복 시간(그림 1의 R : round trip time)동안 자신이 수신자에게 보낸 데이터 셀의 수(E)를 빼서 자신(송신자)의 새로운 크레딧(그림 1의 Credit\_Count)로 정한다. 여기서 각 가상 채널을 위한 버퍼 크기는 N2구간과 N3 구간을 합친 것으로 N2구간은 미리 정해지고(보통의 경우 10으로 정함, 단위 : 셀), N3구간은 송신자가 수신자로 데이터를 보낸 후 다시 수신자가 송신자에게 크레딧을 전송하여 이 크레딧 셀이 도착할 때까지의 왕복 시간(R)동안 목적 대역폭( $B_{target}$  : target bandwidth)으로 전송하기 위한 셀의 갯수만큼의 구간으로 그 크기는 식 1에서와 같이 주어진다.

$$N3 = R * B_{target} / Cell\_Size \quad (1)$$

따라서, 링크의 왕복 시간이나 가상 채널의 목적 대역폭이 증가함에 따라 N3구간이 증가하는 단점이 있다.

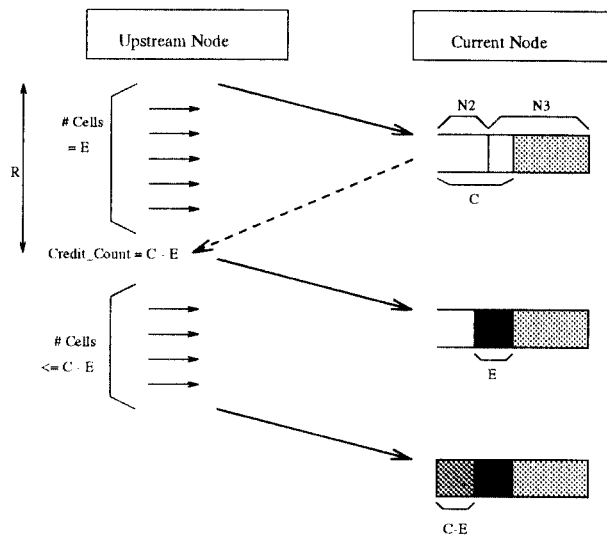


그림 1. N23 기법

N23기법은 대역폭 이용율을 극대화하고 버퍼 오버플로우를 일으키지 않는다는 점에서는 좋으나 항상 각 가상 채널에 대해 N23구간만큼의 버퍼를 확보하고 관리해야 하는 단점이 있다. 또한 각 링크 단위로 흐름 제어하고 또 가상 채널마다 크레딧 및 버퍼 관리를 하기 때문에 노드마다 처리 부하가 크다는 점도 간과할 수 없다.

2.2 CLRC<sup>(7,8)</sup>

모토롤라의 Hluchyj가 제안한 CLRC(Closed-Loop Rate Control)기법은 종점간 전송율을 제어하는 기법으로서 그림 2에서와 같이 송신자의 전송율을 네트워크의 상황에 따라 증감시키는 방법으로 중간 노드에서 큐의 길이가 일정한 임계점을 초과했는지의 여부에 따라 혼잡 상태를 나타내는 명시적 순방향 혼잡 지시(EFCI : Explicit Forward Congestion Indication)를 매번 전송되는 ATM 셀 헤더의 PTI(Payload Type Indicator) 필드를 이용하여 나타내고, 이 데이터 셀이 수신자에게 도착하면 수신자는 현재 네트워크의 상태(즉 혼잡 여부)가 바뀔 때마다 송신자에게 이를 알리는 제어 셀(control cell)을 보내어 송신자가 이를 받아 전송율을 증가(식 2) 내지는 감소(식 3)시키는 기법이다.

$$R = R + \alpha * SCR \tag{2}$$

$$R = \beta * R \tag{3}$$

시뮬레이션에서는 프레임 릴레이의 권고안과 같이  $\alpha$ 로는 1/16,  $\beta$ 로는 0.875, 그리고 최선형 트래픽은 대부분의 경우 유지가능 전송율(SCR : Sustainable Cell Rate<sup>(11)</sup>)이 0이므로 SCR대신 링크의 전체 대역폭에서 보장형 트래픽이 차지하는 평균 대역폭을 뺀 잔여 대역폭( $B_{link} - B_{guaranteed}$ )을 사용했다. 이 기법은 중간 노드들이 평균 큐 길이만 측정하면 되고, 종단 노드(end-to-end node)들만이 전송율을 제어하거나 혼잡 여부를 알리는 등의 역할을 하므로 전체적인 처리 부하가 비교적 작은 장점이 있다. 그러나 전체 연결의 길이가 증가하면 할수록 네트워크의 혼잡 상황 여부에 대한 반응이 늦어지는 단점이 있다. 또한 전송율의 증감이 매우 작은 폭으로 변하므로 네트워크의 보장형 트래픽의 차지하는 대역폭의 변화가 심할 때는 신속하게 전송율을 그에 맞게 조절하지 못하는 단점이 있다.

Ⅲ. 제안된 RCPBA기법

3.1 동작 원리

제안된 RCPBA기법은 네트워크의 중간 노드들이 일정 시간 간격(그림 3의 Tperiod)마다 자신을 지나는

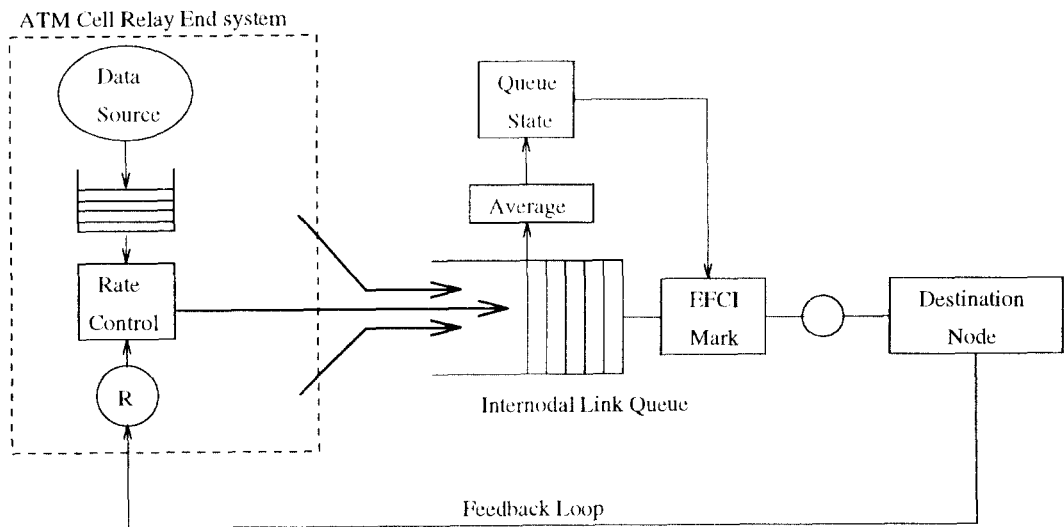


그림 2. CLRC 기법

보장형 트래픽이 차지하는 대역폭을 계산하고 있다. 최선형 가상 채널의 수신자가 보낸 대역폭 할당 셀 (RAC : Rate Allocation Cell)이 지나가면, 중간 노드에서 직전  $T_{period}$  동안의 정보를 이용하여 각 가상 채널에 대해 잔여 대역폭을 할당해 주는 전송을 제어 기법이다. 수신자는  $T_{period}$ 와 같은 주기로 RAC를 보내서 송신자에게 전송을 할당한다. 그러면 송신자는 자신에게 전달된 RAC에 할당된 대역폭에 따라 매 번 전송을 조정하여 최선형 트래픽을 송신한다. 이 기법은 네트워크 노드의 잔여 대역폭을 효율적으로 이용할 수 있는 기법으로 그림 3에서와 같이 동작한다. 여기서 중간 노드들은 각 주기( $T_{period}$ )마다 그 동안의 보장형 트래픽의 평균 대역폭을 측정하는 것으로 정하였고, 수신자도 같은 주기( $T_{period}$ )로 RAC를 송신자에게 전송하는 것으로 정하였다. 이 주기가 길면 길수록 더욱 정확하게 보장형 트래픽의 평균 대역폭을 측정할 수 있지만 너무 크면 측정된 대역폭에 네트워크의 급작스런 변화를 신속하게 송신자에게 알리지 못하여 혼잡 상황에 빨리 대처하지 못하는 단점이 있다. 그리고 실제로 대역폭을 할당할 때는 링크의 전체 대역폭(식 4의  $B_{link}$ )에서 보장형 트래픽의 대역폭( $B_{guaranteed}$ )을 뺀 잔여 대역폭을 각 최선형 트래픽 가상 채널에 균등하게 배분한다. 그리고 잔여 대역폭을 그대로 할당하기보다는 혼잡 상태가 될 확률을

조금이라도 적게 하기 위해 링크 전체 대역폭에 1 미만의 값인  $\alpha$ 를 곱한 대역폭에서 보장형 트래픽의 대역폭을 뺀 대역폭을 최선형 트래픽 가상 채널 수( $n_{best-effort VC}$ )로 나누어서 식 4에서와 같이 할당한다.

$$B_{allocated} = (\alpha * B_{link} - B_{guaranteed}) / n_{best-effort VC} \quad (0 < \alpha \leq 1) \quad (4)$$

### 3.2 장단점 분석

RCPBA 기법은 네트워크의 중간 노드나 송·수신자에게 과도한 처리 부하를 요구하지 않으며 기존의 전송을 제어 기법에 비해 네트워크 노드에서의 대역폭 이용률 등의 성능이 우수하다고 할 수 있다. 특히 외부 보장형 트래픽이 버스티(bursty)하지 않은 경우에는 좋은 성능을 나타낼 수 있는 것이 장점이다. 그리고 역방향으로 전송되는 RAC 셀의 전송 주기가 일정하므로 호 허용 제어시에 역방향의 대역폭을 미리 예약할 수 있다. 그리고 보장형 트래픽이 버스티한 경우에는 주기를 길게 하거나 보장형 트래픽에 대해 Leaky Bucket, Spacer, Moving Window<sup>(9,10)</sup> 등의 사용 인자 제어 방법을 통해 트래픽 셀의 도착 시간의 버스티한 분포를 완화시키는 방법을 사용하면 더 좋은 성능을 나타낼 수 있다.

제안된 기법은 대역폭 할당만 하고 현재 네트워크 노드의 큐 길이를 반영할 수 없다는 것이 단점인데, 이 점

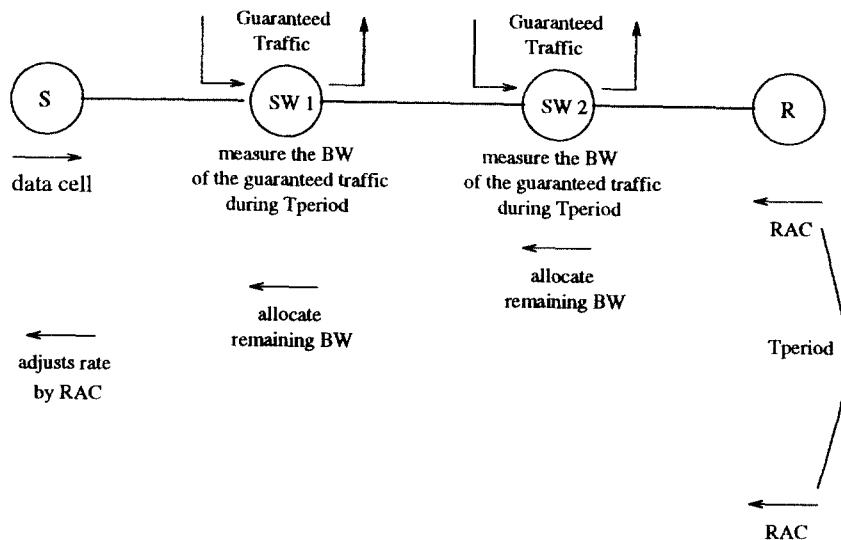


그림 3. RCPBA 기법

은 매번 잔여 대역폭을 계산할 때 그 순간에 큐에 대기하고 있는 셀들을 고려하면 된다. 즉 식 5에서와 같은 기간 동안 도착한 보장형 트래픽의 셀 수(식 5의  $N_{guaranteed}$ )에 잔여 대역폭 계산 순간에 큐에 대기하고 있는 셀의 수(식 5의  $N_{queue}$ )를 더하여 그 수에 셀 하나를 처리하는(링크로 전송하는) 시간(식 5의  $T_{transmit}$ )을 곱한 것이 바로 다음  $T_{period}$ 기간 동안 큐에 대기하고 있던 셀들을 처리하는 시간에 다음 기간에 예상되는 보장형 트래픽 셀들이 차지할 것으로 예상되는 시간의 합이 된다. 따라서 이 시간( $(N_{queue} + N_{guaranteed}) * T_{transmit}$ )이 전체 기간( $T_{period}$ )에서 차지한 비율을  $\alpha$ 에서 뺀 값에 그 노드의 링크 대역폭( $B_{link}$ )을 곱한 것을 현재 전송 중인 최선형 가상 채널에 할당되는 대역폭이 된다. 이렇게 최선형 가상 채널에 할당되는 대역폭에 대한 최종적인 식은 다음과 같다.

$$B_{allocated} = (\alpha - (N_{queue} + N_{guaranteed}) * T_{transmit} / T_{period}) * B_{link} / \Omega_{best-effort VC} \quad (0 < \alpha \leq 1) \quad (5)$$

3.3 버퍼 크기

네트워크의 중간 노드에서 버퍼 오버플로우를 방지하기 위한 버퍼의 크기는 최악의 경우를 고려하면 다음과

같이 계산된다. RCPBA에서 버퍼 크기가 최대로 필요할 때는 다음 그림 4에서와 같이 한 구간에서는 보장형 트래픽의 셀의 도착이 없어서 최선형 트래픽의 가상 채널에 노드의 전체 링크 대역폭을 할당했는데(정확히는 전체 대역폭의 만큼), 그 다음 구간동안 계속해서 보장형 트래픽이 자신의 피크 전송율( $R_{guaranteed}^{peak}$ )로 셀로 전송하는 경우이다.

이 때 필요한 버퍼 크기(식 6의  $S_{buffer}$ )는 식 6에서와 같이 계산된다.

$$S_{buffer} = T_{period} * ((R_{guaranteed}^{peak} + \alpha * B_{link}) - B_{link}) \quad (6)$$

$$= T_{period} * (R_{guaranteed}^{peak} + (\alpha - 1) * B_{link})$$

그런데, 만약 보장형 트래픽이 사용 인자 제어 기법(9.10)을 사용하여 일정 기간(그림 5에서  $T_{interval}$ ) 내에서는 평균 전송율( $R_{average}$ ) 이상 전송 못하도록 제어할 수 있다면 오버플로우를 방지하는데 필요한 버퍼 크기는 식 6보다 더욱 감소될 수 있다. 즉 보장형 트래픽이  $T_{interval}$ 내에서는 평균 전송율( $R_{average}$ ) 이하로만 전송이 되고 피크 전송율( $R_{peak}$ )으로는 피크 유지기간( $T_{peak}$ ) 이내로만 전송이 가능하다고 하면(여기서  $T_{peak} = T_{interval} * R_{average} / R_{peak}$ 라고 생각한다), 가장 최악의 경우는 그림 5에서와 같이 처음 구간에서는 보장형 트래

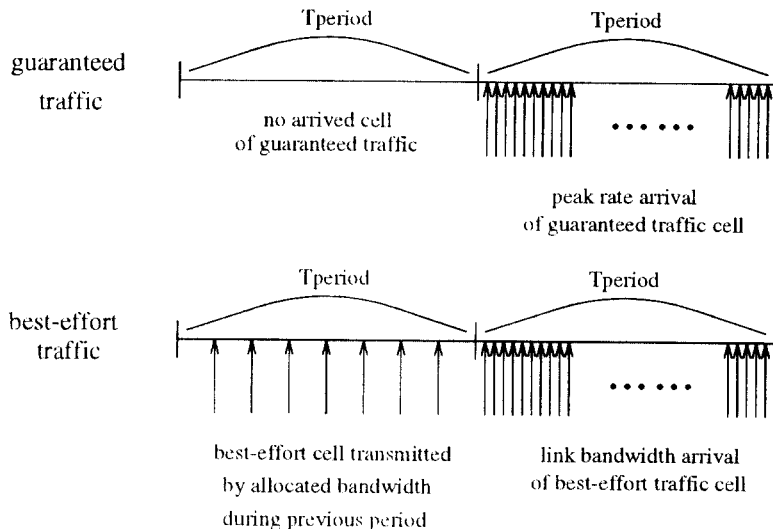


그림 4. 최악의 경우

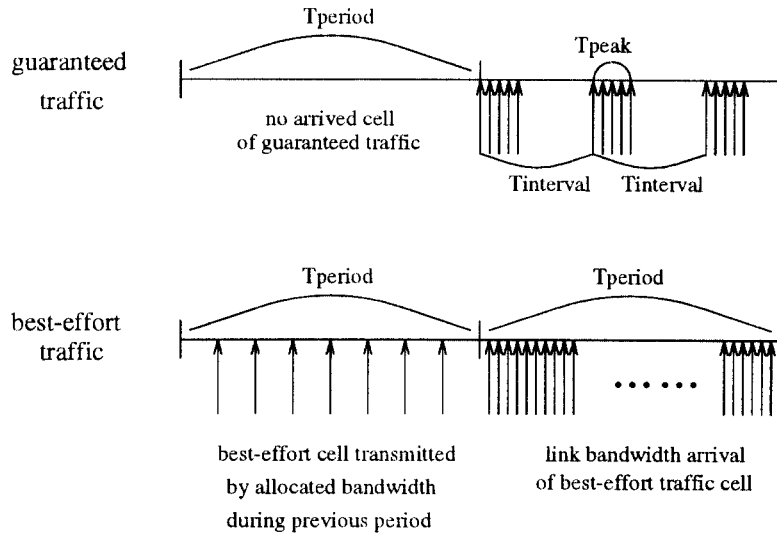


그림 5. 보장형 트래픽 제어시 최악의 경우

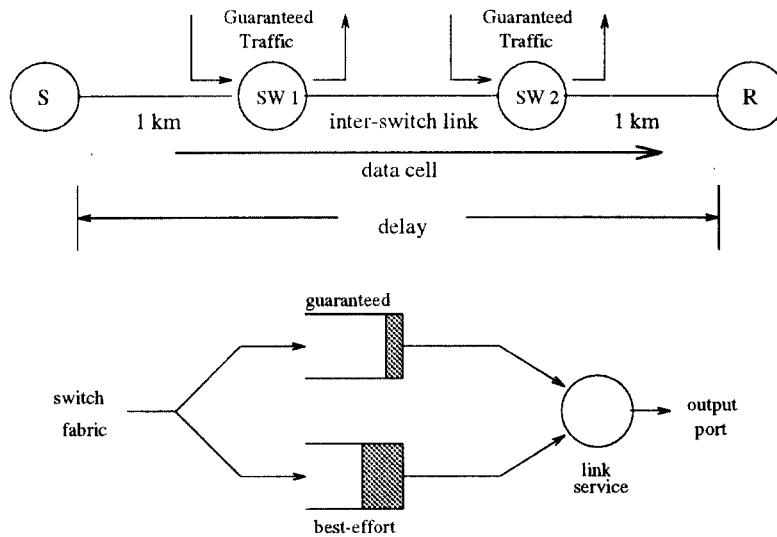


그림 6. 시뮬레이션 환경

픽이 도착하지 않고, 다음 구간에서는 사용 파라미터가 허용하는 한 최대로 대역폭을 차지하는 경우로 이 때 보장형 트래픽과 최선형 트래픽 셀을 오버플로우하지 않기 위한 버퍼 크기는 식 7에서와 같다.

$$S_{reduced\_buffer} = T_{interval} * R_{average} \lfloor T_{period} / T_{interval} \rfloor$$

$$+ R_{peak} * \min(T_{peak}, (T_{period} - T_{interval} * \lfloor T_{period} / T_{interval} \rfloor)) \tag{7}$$

$$+ T_{period} * (\alpha * B_{link} - B_{link})$$

3.4 대역폭 분배시 고려 사항

식 4와 5에서는 최선형 가상 채널들이 동일하게 대역

폭을 분배받게 되나 만약에 최선형 트래픽의 연결 설정 시 대역폭을 요구할 수 있도록 하는 경우에는 식 4와 5에서의  $\frac{1}{n_{best-effort VC}}$ 은 아래 식 8로 대체된다. 즉 식 8은 현재 최선형 가상 채널들의 요구하는 대역폭의 합친 것 (식 (8)의  $\sum_{i \in S} B_i$ ) 중에서 이 RAC가 속한 가상 채널의 요구 대역폭( $B_{this VC}$ )이 차지하는 비율이 된다. 식 8의 S는 현재 전송 중인 최선형 가상 채널의 집합이다.

$$\frac{B_{this VC}}{\sum_{i \in S} B_i} \quad (8)$$

## IV. 시뮬레이션

### 4.1 시뮬레이션 환경

본 논문에서는 그림 6에서 나타난 바와 같이 토폴로지는 탠덤(tandem)으로 연결된 4개의 노드에서 양쪽 종단에 송신자와 수신자를 설정하였고, 중간 노드에서 보장형 트래픽의 도착 시간 분포가 대부분의 경우 포아송 프로세스를 따르는 것으로 설정하였다. 또 보장형 트래픽이 평균적으로 차지하는 대역폭을 다양하게 설정하였고, 그리고 중간 노드간의 길이(그림 6의 inter-switch link)도 변화를 주면서 시뮬레이션 실험을 하였다.

각 링크의 전체 대역폭은 모두 155Mbps으로 설정하였고, 보장형 트래픽은 최선형 트래픽의 데이터 셀보다 서비스 우선 순위를 높게 할당하였다. 최선형 트래픽의 송신자는 흐름 제어 기법이 허용하는 한 즉시 데이터 셀을 발생시키도록 하였고, 수신자가 데이터 셀을 처리하는 시간은 무시하였다. 또한 제어 셀은 큐잉 지연이 없이 바로 링크 서비스를 받는 것으로 설정하였다.

### 4.2 성능 평가 인자

최선형 트래픽을 위한 흐름 제어 기법에서는 중간 노드의 대역폭 이용률도 중요하지만 트래픽을 적절하게 전송하는 것도 중요하다. 즉 흐름 제어 기법의 성능 평가 기준으로는 중간 노드의 대역폭 이용률과 평균 큐 길이, 종점간 지연 시간도 중요하지만, 처리율을 지연 시간으로 나눈 값인 파워(power)가 매우 중요하다고 할 수 있다. 왜냐하면 파워 값이 최대일 때가 가장 최적으로 전송을 하는 때라고 볼 수 있기 때문이다. 따라서 시뮬레이션 결과도 다양한 시뮬레이션 환경에서 각 기법의

제어 인자 값의 변화에 따라 각각의 흐름 제어 기법이 어떠한 성능을 나타내는지, 또 어떠한 경우에 최대의 파워 값을 갖는 지에 초점을 맞추었다.

### 4.3 N23 기법의 제어 인자

N23기법에서는 메카니즘이 비교적 단순하기 때문에 메카니즘에서는 변경시킬 부분이 없었고, N2값과 버퍼 크기(N2+N3 혹은 N3+AdditiveCell(가산 셀))를 성능에 영향을 끼치는 제어 인자로 간주하고 각각의 시뮬레이션 환경에서 적합한 성능을 나타내는 값을 구해 보았다.

### 4.4 CLRC 기법의 제어 인자

CLRC기법은 메카니즘의 변경시킬 부분으로 크게 네 크워크의 노드에서 평균 큐 길이 측정 방법 및 수신자에서 도착하는 각각의 ATM셀 헤더 정보를 갖고서 어떻게 혼잡 여부를 결정한 것인가 즉 여과(filtering) 방법이 있었고, 제어 인자로는 혼잡 여부를 결정하는 임계점, 네트워크의 혼잡 상태에 대한 정보를 받았을 때 송신자가 전송율을 어느 정도 변경시킬 것인가(식 2.3에서  $\alpha$ 와  $\beta$ ), 그리고 왕복 시간 동안 몇 번이나 전송율을 조정할 것인가(조정율)가 성능에 변화를 주는 인자로 간주되었다.

### 4.5 RCPBA 기법의 제어 인자

RCPBA기법은 목적 대역폭 이용율인  $\alpha$ 값과  $T_{period}$ 값이 가장 성능에 영향을 끼치는 인자이고 또한 보장형 트래픽이 평균 전송율이 사용 파라미터 제어되는 기간(3.2절의  $T_{interval}$ )도 제어 인자가 될 수 있을 것이다.

## V. 결과 및 분석

본 장은 각각의 기법이 제어 인자의 변화나 시뮬레이션 환경의 변화에 따라 어떤 시뮬레이션 결과를 나타내는 지에 대한 내용이다. RCPBA기법은 비교적 거리에 무관한 성능을 보였고 N23기법은 거리가 길어질수록 좋은 성능을, CLRC기법은 거리가 길수록 낮아지는 성능을 보였다. 보장형 트래픽의 평균 대역폭이 증가할수록 세 기법 모두 대역폭 이용율이 낮아지는 결과를 보였다.

세 기법의 성능 결과를 비교했을 때는 RCPBA기법이



대역폭 이용률과 파워는 매우 높고 평균 큐 길이와 지연 시간은 비교적 작았다. 참고로 시뮬레이션 결과에서 시간의 단위는 마이크로초( $\mu s$ ), 큐 길이 단위는 셀이며, 파워의 단위는  $cell/\mu s^2$ 이다. 그리고 그래프에 나타난 수치는 대부분 신뢰 구간 95%, 오차율은 5%이하로 구한 평균값임을 밝힌다.

5.1 N23기법 결과

실험 1 : 외부 보장형 트래픽이 링크 대역폭의 50%를 차지할 때 N2값이 증가함에 따라 중간노드의 대역폭 이용률, 평균 큐 길이, 지연 시간을 측정 한 결과가 각각 그림 7, 8, 9이다. N2값이 커짐에 따라 버퍼가 증가하므로 대역폭 이용률, 평균 큐 길이, 지연 시간 모두 증가하는 것을 볼 수 있다. 또한 그림 7에서 중간 노드간 길이가 증가함에 따라 N3구간이 커지므로 중간 노드간 길이가 증가함에 따라 버퍼 크기가 커지고 대역폭 이용률이 증가하는 것을 알 수 있다. 즉 그림 7에서 버퍼 크기만 크게 해 주면 대역폭 이용률은 증가한다는 것을 알 수 있다.

실험 2 : 실험 1의 결과에 따르면 N2(즉 버퍼)를 크게만 해 주면 대역폭 이용률은 증가하지만 지연 시간도

같이 증가하기 때문에 적합하게 전송한다고 볼 수 없다. 따라서 어느 경우에 적합하게 전송하는 지 알아보기 위해 N2값의 변화에 따른 파워(처리율/지연시간) 값을 구한 결과가 그림 10에 나타나 있다. 전반적으로 N2가 작을수록 파워가 큰 것을 알 수 있다.

실험 3 : 실험 2는 N2값이 변하면 버퍼 크기도 변하지만 크레딧 셀을 변하는 주기도 변하기 때문에 언제가 적합한 버퍼 크기인지 정확하다고 할 수 없었다. 따라서 이번에는 크레딧 전송 주기로서의 N2값을 1로 고정시키고 N3구간에 추가로 버퍼(그림 11의 *additive cell*)를 할당하여 버퍼 크기가 언제 최대인지 알아보는 실험을 하였다. 여기서 보장형 트래픽의 대역폭은 링크 대역폭의 50%를 차지한다. 그 결과 그림 11에서 나타난 바와 같이 거의 N3구간에 가까울 때에 파워가 최대로 되는 것을 알 수 있었다. 따라서 역방향 대역폭이 문제가 되지 않는다면 가산 셀의 크기와 N2를 줄였을 때에 적합한 성능을 나타낼 수 있다고 하겠다.

실험 4 : 이 실험은 보장형 트래픽의 평균 대역폭이 변함에 따라 중간 노드의 대역폭 이용률의 변화를 나타낸 것으로 보장형 트래픽의 대역폭이 클수록 목적 대역폭(식 1의  $B_{target}$ )이 줄고 따라서 최선형 트래픽의 버퍼

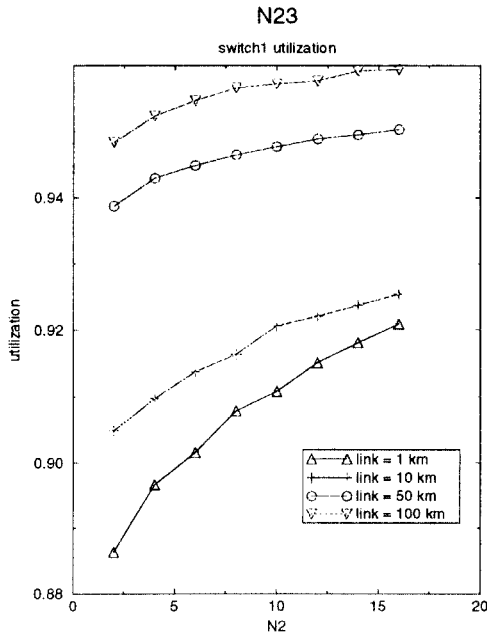


그림 7. N2값에 따른 대역폭 이용률

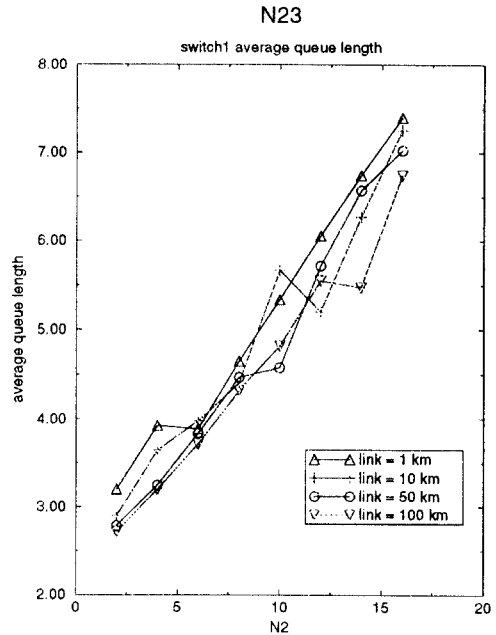


그림 8. N2값에 따른 평균 큐 길이

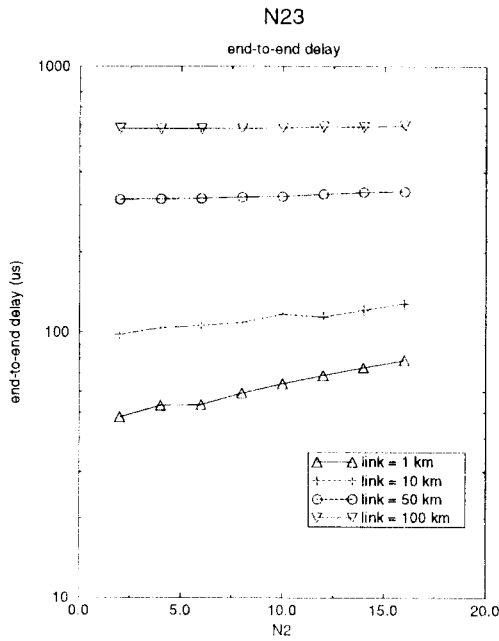


그림 9. N2값에 따른 지연 시간

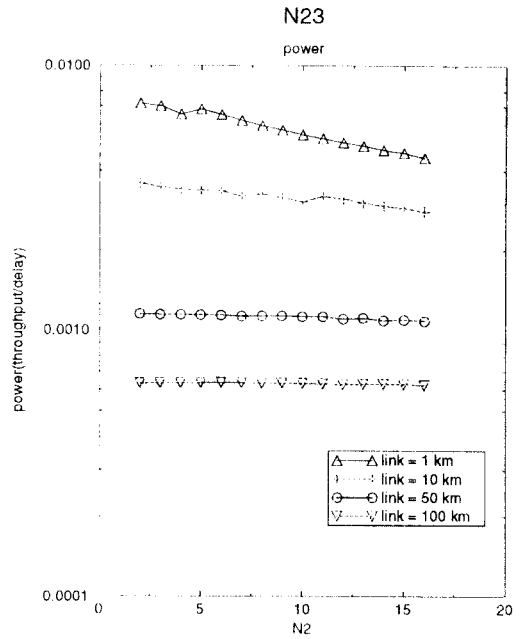


그림 10. N2값에 따른 파워

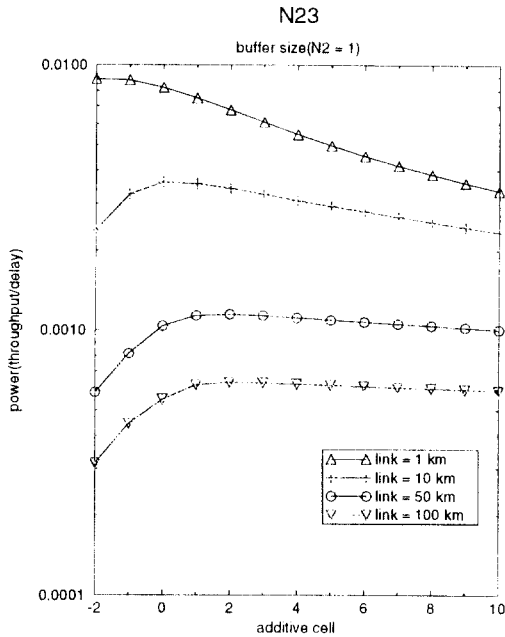


그림 11. 버퍼값에 따른 파워

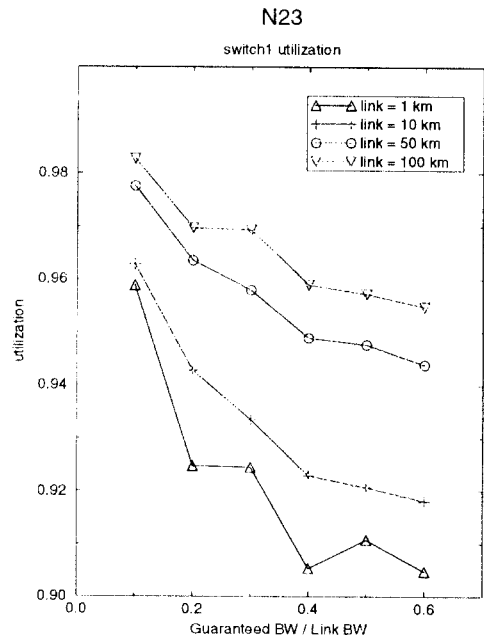


그림 12. 보장형 트래픽의 대역폭에 따른 이용율

크기가 작아지므로 그림 12에서와 같이 대역폭 이용율이 감소하는 것을 볼 수 있다.

5.2 CLRC기법

실험 5 : 임계점이 증가함에 따라 대역폭 이용율, 평균 큐 길이, 지연 시간의 변화를 각각 그림 13, 14, 15에서와 같이 나타내었다. 임계점 값이 커짐에 따라 혼잡을 파악하는 것이 둔해지므로 전반적으로 전송율이 커져서 대역폭 이용율, 평균 큐 길이, 지연 시간 모두 증가한다는 것을 알 수 있다.

실험 6 : 이 실험은 임계점의 값이 변할 때 파워값이 언제 최대치를 나타내는 지를 알기 위한 것으로 그림 16에 나타난 것을 보면 전반적으로 중간 노드간의 길이가 길어짐에 따라 임계점의 값을 높이는 것이 성능을 최적화하는 방법임을 알 수 있다.

실험 7 : 전송율 조절율(fraction) 즉, 한 왕복 시간 동안에 몇 번이나 전송율을 조절할 것인가를 변경시키면서 파워의 변화를 알아보는 실험이다. 그림 17에서와 같이 왕복 시간이 짧은 경우에는 조절율이 증가함에 따라 파워가 더욱 급격히 감소함을 알 수 있으며 전반적으로 조절율이 작을 수록 파워가 큰 결과가 나왔다.

실험 8 : 이 실험은 전송율 조절시 변경되는 폭을 나타내는  $\alpha$ ,  $\beta$ 의 값이 과연 프레임 릴레이의 권고안대로 따르는 것이 좋은가에 대한 것이다.  $\alpha$ 값은 중간 노드간 길이가 증가함에 따라 크게 해 주는 것이 성능을 향상시킬 수 있다는 것이 그림 18에 나타나 있다. 그리고  $\beta$ 값은 전반적으로 0.95이상일 때 파워가 좋게 나타나서 보장형 트래픽이 포아송 분포일 때는  $\beta$ 값을 권고안보다 크게 설정하는 것이 좋다는 결과가 그림 19에 나타나 있다.

실험 9 : 보장형 트래픽의 평균 대역폭 변화시 중간 노드의 전체 대역폭 이용율을 알아보는 실험이다. 그림 20의 결과를 보면 외부 보장형 트래픽의 대역폭이 증가함에 따라 식 2의 SCR에 해당하는 잔여 대역폭이 감소하므로 전송율을 증가시키는 양이 감소하는 데에 원인이 있는 것으로 분석된다.

5.3 RCPBA기법

실험 10 : 이 실험은  $\alpha$ 값의 변화에 따른 대역폭 이용율, 평균 큐 길이, 지연 시간의 변화를 알아보기 위한 것으로 각각 그림 21, 22, 23에 결과가 나타나 있다.  $\alpha$  값이 커짐에 따라 송신자에게 할당되는 전송율이 증가하

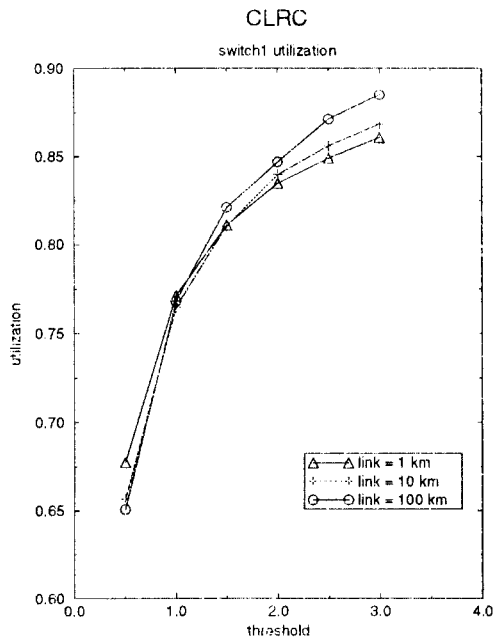


그림 13. 임계점값에 따른 대역폭 이용율

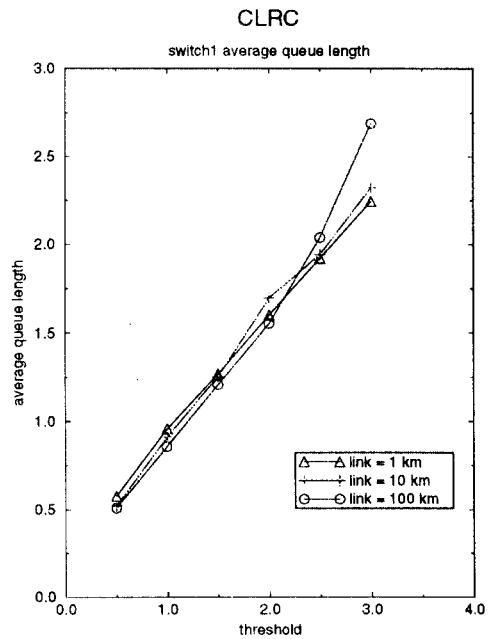


그림 14. 임계점값에 따른 평균 큐 길이

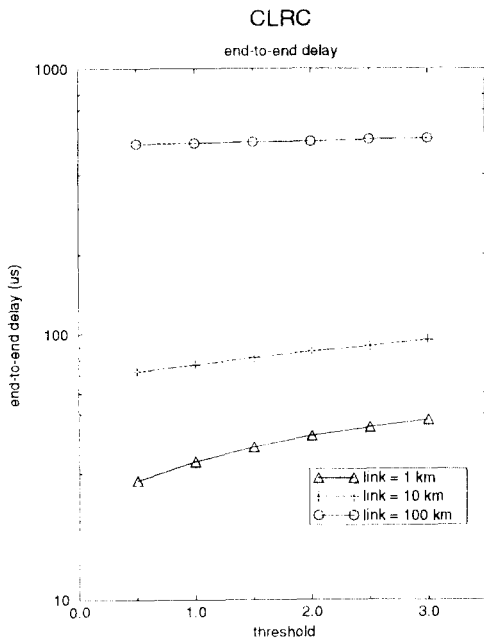


그림 15. 임계값에 따른 지연 시간

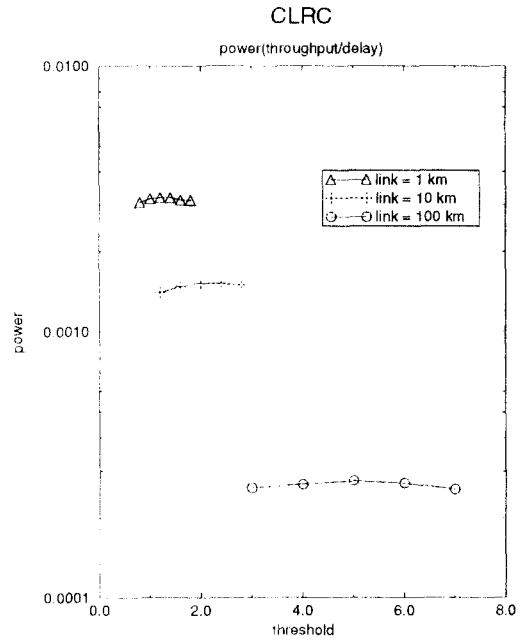


그림 16. 임계값에 따른 파워

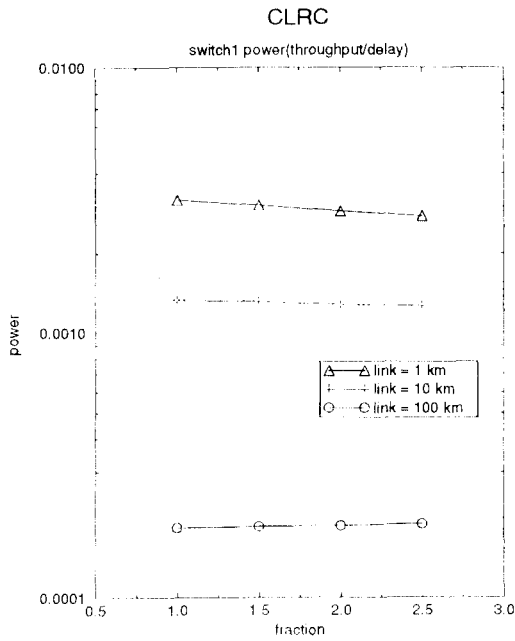


그림 17. 조정율에 따른 파워

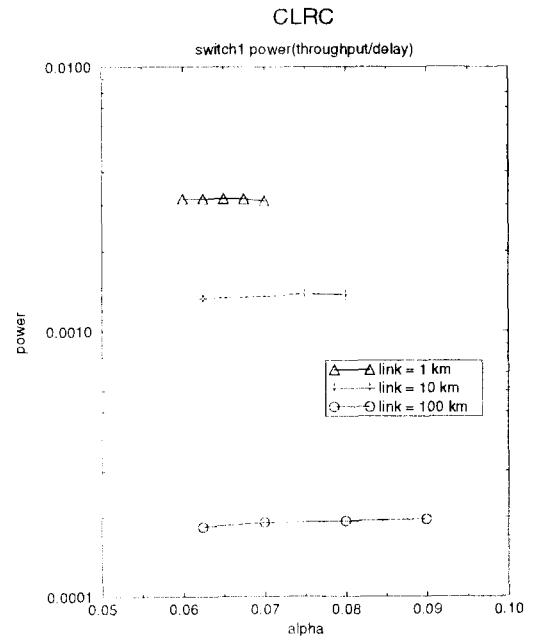


그림 18.  $\alpha$ 에 따른 파워

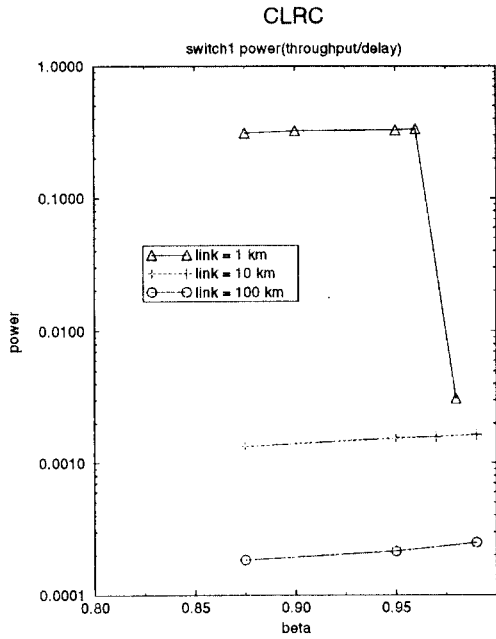


그림 19.  $\beta$ 값에 따른 파워

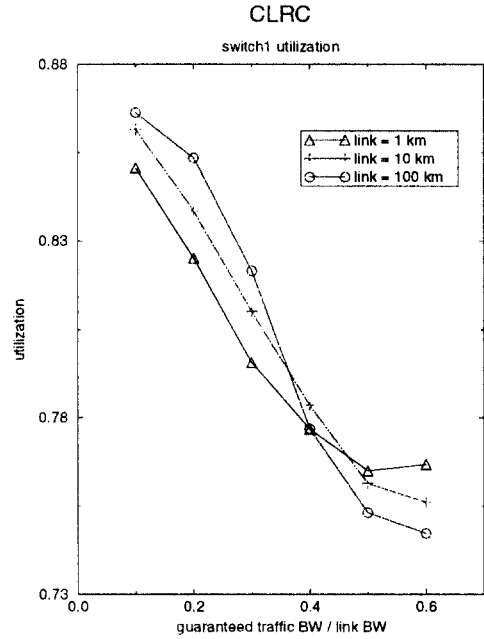


그림 20. 보장형 트래픽의 대역폭에 따른 대역폭 이용률

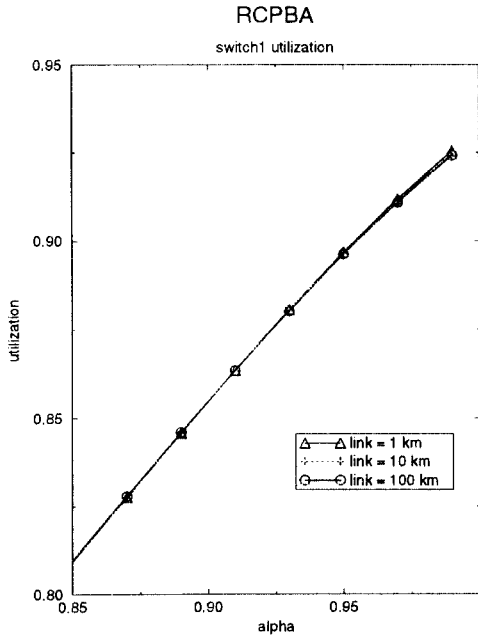


그림 21.  $\alpha$ 값에 따른 대역폭 이용률

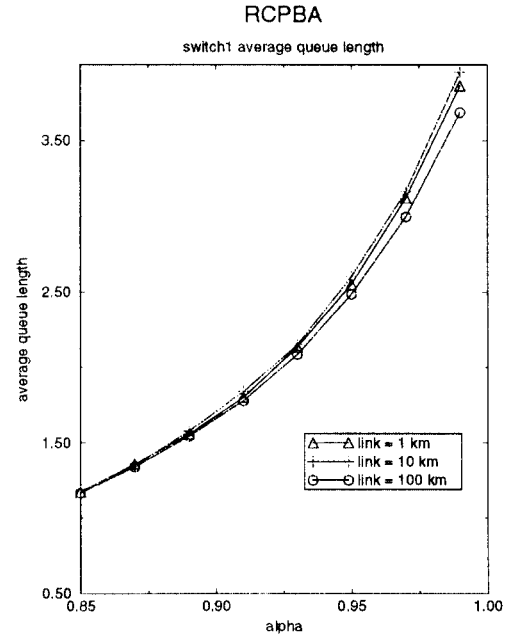


그림 22.  $\alpha$ 값에 따른 평균 큐 길이

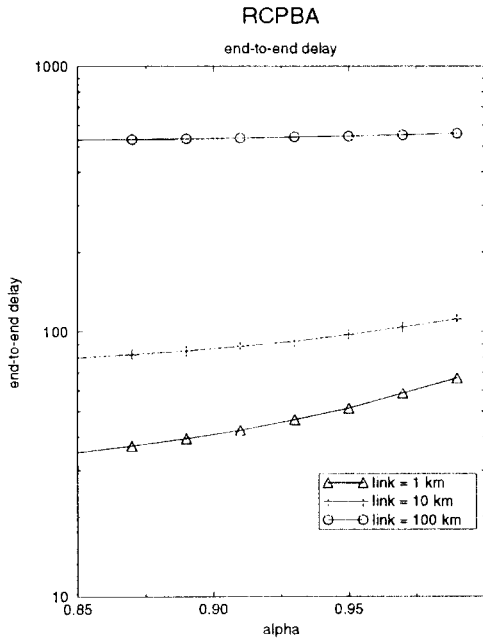


그림 23.  $\alpha$ 값에 따른 지연 시간

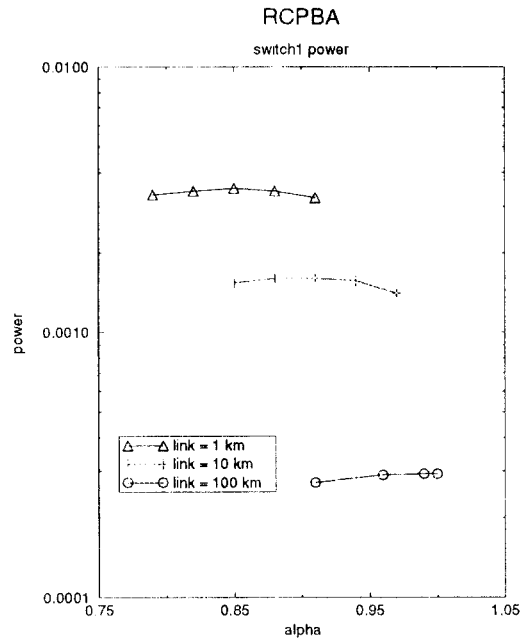


그림 24.  $\alpha$ 값에 따른 파워

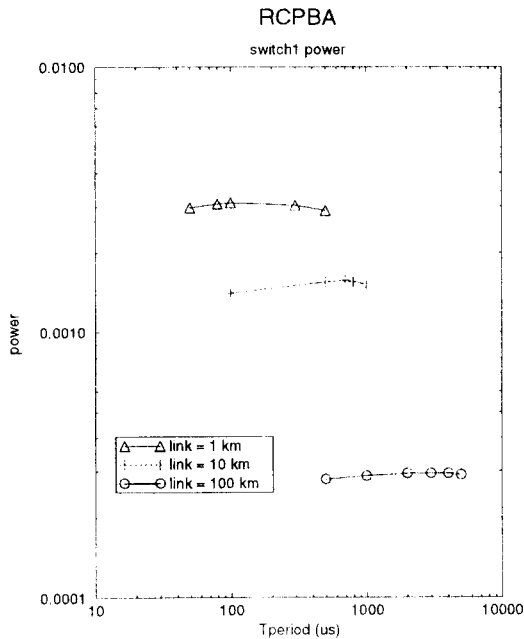


그림 25.  $T_{period}$ 값에 따른 파워

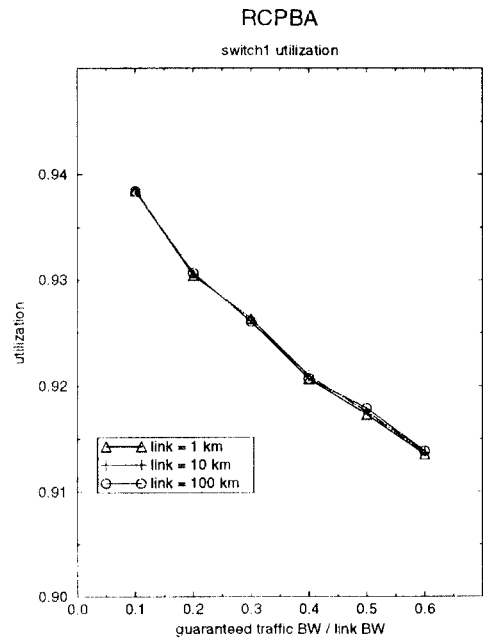


그림 26. 보장형 트래픽의 대역폭에 따른 대역폭 이용률

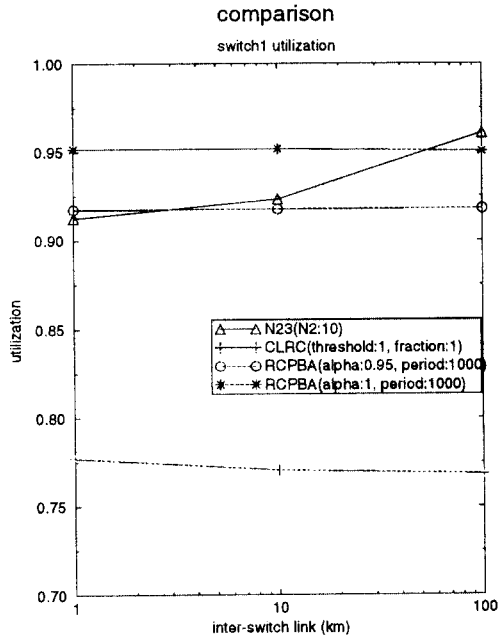


그림 27. 대역폭 이용률의 비교

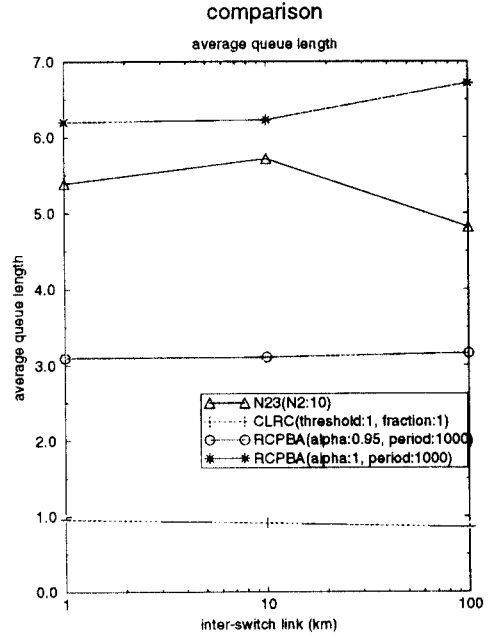


그림 28. 평균 큐 길이의 비교

므로 대역폭 이용률, 평균 큐 길이, 지연 시간 모두 증가함을 알 수 있다.

실험 11 : 각 상황에서 값의 변화에 따른 최적의 파워값을 찾기 위한 실험이다. 중간 노드간 길이가 길수록 값을 크게 하는 것이 적합한 성능을 나타낸다는 결과가 그림 24에 나와 있다.

실험 12 :  $T_{period}$  값의 변화에 따른 파워값의 최대치 구간을 찾기 위한 실험이다. 중간 노드간 길이가 길수록  $T_{period}$ 를 길게 하는 것이 성능을 최적화하는 것으로 그림 25에 나타났다. 즉  $T_{period}$ 를 송 수신자간의 지연 시간의 수 배로 정하는 것이 적합한 성능을 나타낸다는 결과가 나왔다.

실험 13 : 보장형 트래픽의 평균 대역폭 변화시 대역폭 이용률의 변화를 알아 보는 실험으로 보장형 트래픽의 평균 대역폭이 증가함에 따라 중간 노드의 대역폭 이용률이 감소하는 결과가 그림 26에 나타나 있다.

#### 5.4 각 기법들간의 비교

실험 14 : 보장형 트래픽이 링크 대역폭의 50%를 차지할 때 대역폭 이용률을 비교하는 결과가 그림 27에

나타나 있다. N23기법은 윈도우 기법이기 때문에 일단 윈도우 크기 만큼의 데이터를 전송하고 또한 대역폭의 저효율을 방지하는 목적이 윈도우 기법의 목적이므로 아무래도 대역폭 이용률이 높게 나온다. 반면에 CLRC기법은 일정 시간 간격으로 셀을 전송하고 전송을 조정시 감소되는 양이 증가하는 양보다 크므로 전반적으로 낮은 대역폭 이용률을 나타낸다. RCPBA기법은 전반적으로 길이에 무관하게 좋은 대역폭 이용률을 나타내고 있다.

실험 15 : 보장형 트래픽이 링크 대역폭의 50%를 차지할 때 평균 큐 길이를 비교하는 결과가 그림 28에 나타나 있다. N23기법은 윈도우 기법이기 때문에 일단 윈도우 크기 만큼의 데이터를 전송하므로 평균 큐 길이가 매우 크다. CLRC기법은 평균 큐 길이를 혼잡 여부로 판별하므로 평균 큐 길이가 임계점 값인 1보다 낮은 결과를 나타낸다. RCPBA기법은 전반적으로 길이에 무관하게 비교적 낮은 평균 큐 길이를 나타내고 있다.

실험 16 : 보장형 트래픽이 링크 대역폭의 50%를 차지할 때 종점간 지연 시간을 비교하는 결과가 그림 29에 나타나 있다. N23기법은 평균 큐 길이가 매우 큰 만큼 지연 시간도 매우 크다. CLRC기법은 가장 낮은

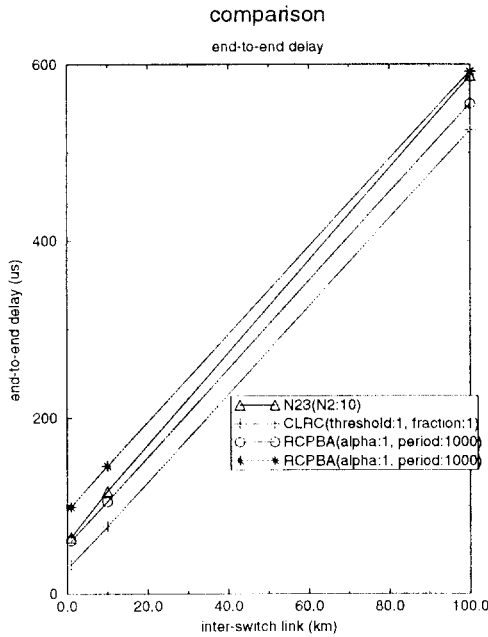


그림 29. 지연 시간의 비교

큐 길이, 가장 낮은 지연 시간을 나타낸다. RCPBA기법은 전반적으로 비교적 낮은 지연 시간을 나타낸다.

실험 17 : 중간 노드간 길이가 변할 때 각 기법의 파워가 어떻게 변하는가를 나타내는 결과가 표 1에 나타나 있다. 전반적으로 N23기법은 길이가 길어질수록 파워가 다른 기법보다 크고, CLRC기법은 길이가 짧을수록 파워가 다른 기법보다 작은 경향을 가진다. RCPBA기법은 길이에 무관하게 비교적 높은 파워 값을 나타냈다.

VI. 결 론

본 논문에서는 주기적 대역폭 할당을 통한 전송을 제어 기법을 제시하고 최근에 ATM망에서 관심이 모아지고 있는 N23, CLRC 흐름 제어 기법과 함께 시뮬레이션 실험을 하였다. 제안된 RCPBA기법에서는 수신자가 주기적으로 대역폭 할당 셀을 송신자에게 보내면 네트워크의 중간 노드들은 이 셀에 허용가능한 대역폭 할당을 표시하고, 송신자는 이 셀에 표시된 대역폭에 맞추어 전송율을 조정한다. 이 기법은 중간 노드 및 종단 노드에 부하가 적으면서도 네트워크 노드의 대역폭을 효과적으로 이용하는 장점을 가졌고 또한 오버플로우를 방지할

표 1. 파워 값의 비교  
Table 1. Comparison of Power

중간 노드간 길이	1 km	10 km	100 km
N23	0.00246	0.00138	0.000300
CLRC	0.00317	0.00133	0.000184
RCPBA	0.00264	0.00152	0.000288

수 있는 버퍼 크기도 간단하게 계산된다. 시뮬레이션 환경은 포아송 분포를 따르는 보장형 트래픽이 차지하고 남은 잔여 대역폭을 어느 흐름 제어 기법이 얼마나 효율적으로 이용하여 최선형 트래픽을 전송할 수 있는가를 파악하는 데에 중점을 두었다. 시뮬레이션 상황의 변화에서는 RCPBA기법은 비교적 중간 노드간 거리에 무관한 성능을 보였고 N23기법은 거리가 길어질수록 좋은 성능을, CLRC기법은 거리가 길수록 낮아지는 성능을 보였다. 그리고 보장형 트래픽의 평균 대역폭이 증가할수록 세 기법 모두 대역폭 이용율이 낮아지는 결과를 보였다. 세 기법의 성능 결과를 비교했을 때는 RCPBA기법이 대역폭 이용율과 파워는 매우 높고 평균 큐 길이와 지연 시간은 비교적 크지 않아서 가장 효율적으로 데이터를 전송하는 기법이라는 결론이 나왔다.

참고문헌

1. H. Armbrüster, and K. Wimmer, "Broadband Multimedia Applications Using ATM Networks : High-Performance Computing, High-Capacity Storage, and High Speed Communication," *IEEE J. Select. Areas in Commun.*, vol. 10, no. 9, pp.1382-1396, Dec. 1992.
2. J. J. Bae and T. Suda, "Survey of the Traffic Control Schemes and Protocols in ATM Network," *Proc. of the IEEE*, vol. 79, no. 2, pp.170-189, Feb. 1991.
3. ITU-T, I.371 : Traffic Control and Congestion Control in B-ISDN, *ITU-T Recommendation*, Mar. 1993.
4. J. W. Roberts, "Traffic Control in the B-ISDN," *Computer Networks and ISDN Systems*, vol. 25, no. 10, pp.1055-1064, 1993.



5. H. T. Kung and A. Chapman, "The FCVC(Flow-Controlled Virtual Channels) Proposal for ATM Networks," *Intern. Conf. on Network Protocols*, pp.116-127, 1993.
6. H. T. Kung and A. Chapman, Adaptive Credit Allocation for Flow-Controlled VCs, *ATM Forum Contribution*, no. 94-0282, March, 1994.
7. N. Yin and M. G. Hluchyj, "On Closed-Loop Rate Control for ATM Cell Relay Networks," *INFOCOM '94*, pp.99-108, 1994.
8. N. Yin and M. G. Hluchyj, "A Dynamic Rate Control Mechanism for Source Coded Traffic in a Fast Packet Network," *IEEE J. Select. Areas in Commun.*, vol. 9, no. 7, pp.1003-1012, Sep. 1991.
9. E. P. Rathgeb, "Modeling and Performance Comparison of Policing Mechanisms for ATM Networks," *IEEE J. Select. Areas in Commun.*, vol. 9, no. 3, pp.325-334, April, 1991.
10. 이우섭, 최대우, 박홍식, "ATM 트래픽에 대한 사용 파라미터 제어 방법들의 고찰," *ETRI 주간 기술 동향 92-25*, July, 1992.
11. ATM Forum, *ATM User-Network Interface Specification*, Prentice Hall, 1993.



權泰境(Kwon Tac Kyoung) 정회원

1971년 1월 31일생  
 1989년~1993년 : 서울대학교 컴퓨터공학과 학사  
 1993년~1995년 : 서울대학교 컴퓨터공학과 석사  
 1995년~현재 : 서울대학교 컴퓨터공학과 박사과정



崔陽熙(Choi Yang Hee) 정회원

1955년 7월 27일생  
 1971년~1975년 : 서울대학교 공과대학 전자공학과(학사)  
 1975년~1977년 : 한국과학기술원 전기 및 전자공학과(석사)  
 1980년~1984년 : 프랑스 국립 전기통신대학 전산과(공학박사)  
 1977년~1979년 : 한국전기통신연구소 연구원  
 1981년~1984년 : 프랑스 국립 전기통신연구소 연구원 (Centre National d'Etudes des Telecommunications)  
 1988년~1989년 : IBM Thomas J. Watson Research Center 방문연구원  
 1984년~1991년 : 한국전자통신연구소 책임연구원, 데이터통신 연구실장, 망기술 연구실장, 프로토콜 연구실장, 정보통신 표준연구센터장 역임  
 1991년~현재 : 서울대학교 컴퓨터공학과 부교수  
 서울대학교 컴퓨터네트워크 연구센터 센터장  
 서울대학교 중앙교육연구전산원 부원장