

ATM 망에서 점프계수를 갖는 HOL 알고리즘을 이용한 효율적 버퍼제어기법

正會員 金 賢 哲*, 文 明 龍**, 李 門 浩*

An Efficient Buffer Control Scheme using HOL Algorithm with Jump Coefficient in ATM Networks

HyonChol Kim*, MyungRyong Moon**, Moon Ho Lee* *Regular Members*

요 약

본 논문에서는 기존에 제안된 HOL(Head Of the Line)이나 HOL-PJ(HOL with Priority Jump)방법을 개선하여 간단한 하드웨어 구성을 가지고 저우선순위 트래픽의 QoS(Quality of Service)를 보장하기 위한 HOL-LJ(HOL with Length Jump)방법을 제안한다. 제안된 알고리즘의 성능 분석은 $M/M/1/N$ 큐잉 모델을 사용하여 분석하였다. 제안된 알고리즘을 이용하여 시간 우선순위 제어를 수행하였을 경우, 구현상 간단함에도 HOL-PJ보다 나은 성능을 유지함을 시뮬레이션 결과를 통하여 확인하였다. 그리고 이에 대한 응용으로 손실우선순위와 혼합한 형태를 제안하고 성능분석을 수행하였다. 손실우선순위와 혼합하여 성능을 비교한 결과 PBS(Partial Buffer Sharing)와 혼합한 구조가 이중 버퍼를 사용하는 방법보다 더 많은 이점을 가지고 있음을 알 수 있다.

ABSTRACT

In this paper, we study new HOL-LJ scheme to guarantee the QoS of low priority cell with simple hardware which is not realizable in previously proposed HOL and HOL-PJ. The performance of the proposed scheme is evaluated by analysing an equivalent $M/M/1/N$ queuing system. In case of using the proposed scheme, we confirmed that good performance compared with HOL-PJ through simulation and mathematical analysis. We also applied HOL-LJ algorithm to conventional Dual Buffer and Partial Buffer, and we analyzed the performance through computer simulation.

*전북대학교 정보통신공학과
Dept. of Infor. & Commun. Eng., Chonbuk National Univ.

**전북대학교 대학원 영상정보공학과
論文番號:96258-0824
接受日字:1996年 8月 24日

I. 서 론

광대역 종합정보통신망(B-ISDN)은 음성, 저속 및 고속 데이터, 정지 및 동화상의 전송 서비스, 다중매체 서비스를 수용하는 고속 정보통신망으로 미래의 새로운 형태의 서비스를 융통성 있게 지원한다. 다양한 대역폭과 서비스의 종류에 따라 고정 전송속도 및 가변 전송속도를 가지는 다양한 서비스들의 요구조건을 만족시키기 위해 ITU-T에서는 B-ISDN의 실현방안으로 ATM 전송방식을 채택하였다[1][2]. 이러한 ATM방식은 연결과 대역할당의 독립성을 특징으로 하는 전송방식으로 호 연결요청시 가상채널(Virtual Channel)을 설정하여 정보를 전달하며, 가상채널이 설정될 때마다 사용자의 요구에 따라 전송용량을 할당한다. 따라서 채널별 전송용량이 고정된 동기식(STM) 전송방식에 비해 ATM망은 사용자의 요구에 따라 전송용량을 동적으로 할당하므로 전송 자원 활용 면에서 장점을 가지게 된다. 그러나 ATM망에서는 접속이 허락된 호에 대하여 최고비트율(PBR)의 대역폭을 제공하지 않고, 필요한 만큼의 대역폭을 할당하므로, 트래픽의 예측 불가능한 특성으로 인해 이미 할당된 대역폭을 일시적으로 초과하여 폭주상태에 빠질 우려가 있는데, 특히 동화상이나 고속 데이터와 같은 버스트성이 큰 트래픽이 폭주에 많은 영향을 미친다. 이러한 망의 폭주 상태가 발생할 경우 각 서비스 트래픽의 서비스 요구 품질(QoS)를 만족하지 못하게 되므로 ATM망에서는 망의 폭주에 의한 셀 손실의 최소화와 망 자원 사용의 극대화란 서로 상충되는 목표를 동시에 달성해야 한다. 따라서 사용자가 요구하는 QoS를 충족시키고 망 자원을 효과적으로 공유하기 위해서는 호 설정 단계나 정보 전송단계에서의 적극적인 트래픽 제어가 필수적이다.

ITU-T에서 권고하고 있는 트래픽 제어기술[16] 중 본 논문에서 다루는 우선 순위 제어방법에는 스케줄링에 의한 시간우선순위 제어와 선택적 셀 폐기에 의한 손실우선순위 제어방법이 있다. 시간 우선순위 제어란 지연에 민감한 트래픽에 우선순위를 주어 큐잉 지연시간을 줄이는 방법이고, 손실 우선순위 제어는 적절한 버퍼링을 통해 우선순위가 높은 셀의 버퍼 오버플로우에 의한 손실을 줄이는 방법이다. 본 논문에서는 기존의 시간우선순위제어의 기법인 HOL이나

HOL-PJ방법에서의 저우선순위 셀의 품질저하 및 하드웨어 복잡도의 단점을 보완하는 새로운 알고리즘인 HOL-LJ방법을 제안한다.

본 논문의 구성은 II장에서는 기존의 HOL 방식을 보완한 새로운 HOL-LJ알고리즘을 제안하고 수식적인 분석을 통하여 각 우선순위의 평균지연을 평가하였다. III장에서는 제안한 방법의 시뮬레이션을 통한 성능을 분석하였다. IV장에서는 제안한 HOL-LJ 알고리즘과 기존의 손실우선순위 제어기법과를 혼합한 제어구조를 제시하고, 이에 대한 성능분석을 시행하고, V장에서 결론을 맺는다.

II. 제안한 우선순위제어방법

ATM 망에서의 지연치는 여러 가지 부분에서 발생하게 된다. 그러나 실제적으로 시스템에 영향을 미치는 지연값은 각 노드의 큐에서 큐잉에의한 지연의 값으로 나타낼 수 있다. 그 이유는 전송지연이나 기타 여러 부분에서 발생하는 지연값은 모든 서비스가 공통적으로 겪는 지연값이거나, 제어에 의하여 줄일 수 있는 지연값이 아니기 때문이다. 따라서 우선순위에 관련된 제어기법에서 고우선순위의 서비스는 각 노드의 큐에서 겪는 지연값을 최소한으로 유지될 수 있도록 제어하는 기법이라고 할 수 있다[8][10]. 이러한 제어기법으로는 기존의 HOL, HOL-PJ, HOL-T, MLT(Maximum Laxity Threshold), QLT(Queue Length Threshold) 등이 여러 논문에서 제안되어졌다[9][11][15][17][18].

1. HOL-LJ 방법

제안한 HOL-LJ방법은 시간우선순위제어를 위한 방법으로서 단일 버퍼를 사용하여 시간우선순위에 따라 차등적으로 서비스할 수 있는 방법이다. 특히 적절한 L값의 설정에 의해 우선순위가 낮은 트래픽(비실시간 트래픽)의 QoS에 대한 고려가 가능하다. 이는 아래 그림 1과 같이 표현된다.

그림에서 알 수 있듯이 L값이 클수록 우선 순위가 높은 트래픽에 대한 우선 순위가 더 고려되어 HOL과 비슷해지며, L값이 작을수록 하드웨어 구현이 간단해지고 우선순위가 낮은 트래픽에 대해 고려해 줄 수 있음에 반하여 우선순위가 높은 트래픽에 대해서는

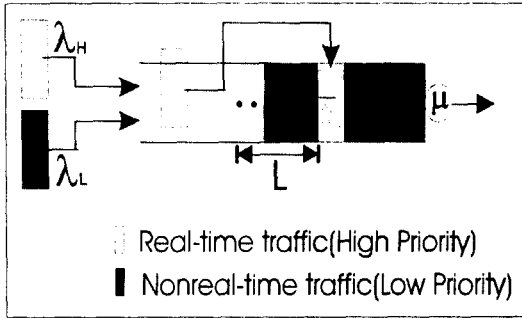


그림 1. HOL-LJ 방법
Fig. 1 HOL-LJ Algorithm

만족스러운 서비스를 하지 못하게 된다. 또한 HOL-LJ 방법을 기존에 제안된 손실우선순위제어 알고리즘과 혼합하여 사용함으로써, 최근의 연구동향인 두 가지 우선순위를 동시에 수행할 수 있다.

HOL-LJ에서 실시간 트래픽의 셀(CELL_R)이 버퍼에 입력되는 경우, 기존에 입력된 비실시간 트래픽 셀(CELL_N)보다 L만큼 앞으로 들어가게 되며, 비실시간 트래픽의 셀이 버퍼에 입력되는 경우에는 버퍼의 가장 마지막 위치에 저장된다. 입력되는 셀의 위치는 식(1), 식(2)에 의해 결정된다.

$$P(CELL_R) = \text{Max}(P(R) + 1, P(N) - L) \quad (1)$$

$$P(CELL_N) = \text{Max}(P(R) + 1, P(N) + 1) \quad (2)$$

표 1. HOL-LJ 알고리즘에 대한 의사코드
Table 1. Pseudo-code of HOL-LJ Algorithm

```

if(셀이 실시간 트래픽인 경우)
then{
  if(max(P(CELLN)-L, P(CELLR)+1) <= BufferSize){
    P(x) = max(P(CELLN)-L, P(CELLR)+1);
    if(P(CELLN)-L = P(CELLR)+1){
      P(CELLN)-L개 만큼의 비실시간 셀위치 1 증가
    }else{
      Real+1 이후의 비실시간 셀의 위치를 1씩 증가
    }
  }
}else{ /* 셀이 비실시간 트래픽인 경우 */
  if(max(P(CELLN)+1, P(CELLR)+1) <= BufferSize)
    P(x) = max(P(CELLN)+1, P(CELLR)+1);
}
    
```

- $P(X)$: 새로운 셀 X의 버퍼에서의 위치,
- $\text{Max}(A, B)$: A와 B중 큰 수를 선택,
- L : 점프 계수(the coefficient of jump),
- R : 실시간 트래픽(real-time traffic),
- N : 비실시간 트래픽(non-real-time traffic).

새로운 셀의 입력이 있을 시 셀의 처리 알고리즘은 표 1에서와 같다.

2. 모델링 및 분석

서비스에 영향을 미치는 지연값의 계산은 각 노드에서 발생하는 지연값을 계산하여 예측할 수 있다. 따라서 제안한 기법의 지연값을 계산하기 위한 모델 파라미터로는 각각의 트래픽의 도착율은 서로 독립적인 관계를 가지는 포아송 분포를 따른다고 정의하고, 버퍼의 모델을 $M/M/1/N$ 모델을 사용하여 버퍼의 상태 천이도를 나타내면 그림 2와 같다. 이와 같은 버퍼 상태천이도의 지연값을 계산하면 노드에서의 각 우선순위별 지연값을 계산할 수 있다.

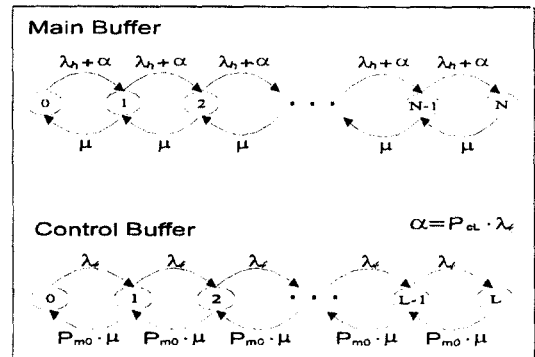


그림 2. 버퍼의 상태 천이도
Fig. 2 The state transition diagram

- 위 상태 천이도에서
- λ_H : 고우선순위의 부하율
- λ_L : 저우선순위의 부하율
- μ : 서비스율
- N : 주 버퍼 크기
- L : 제어부에 있는 버퍼의 크기
- P_{m_i} : 셀이 주버퍼의 i번째 공간에 존재할 확률
- P_{c_i} : 셀이 제어버퍼의 i번째 공간에 존재할 확률

α : 제어버퍼에서의 overflow 확률로 가정하기로 한다.
 위 그림에서 Main 버퍼에 대하여 수식으로 전개하면,

$$P_{m0}(\lambda_h + \alpha) = \mu P_{m1} \quad (3)$$

$$(\lambda_h + \alpha) P_{m0} + \mu P_{m2} = (\lambda_h + \alpha) P_{m1} + \mu P_{m1} \quad (4)$$

와 같이 나타내어진다.

식(3)에서 $P_{m1} = \frac{\lambda_h + \alpha}{\mu} P_{m0}$ 을 구할 수 있다. P_{mk} 의 식을 P_{m0} 와 λ_h, α, μ 로 표현하면, 일반화식 $(\lambda_h + \alpha) P_{m(i-1)} + \mu P_{m(i+1)} = (\lambda_h + \alpha) P_{mi} + \mu P_{m(i+1)}$ 을 얻을 수 있다.

정리하면

$$\begin{cases} \frac{P_{mi} - P_{m(i-1)}}{P_{m(i+1)} - P_{mi}} = \frac{\mu}{\lambda_h + \alpha}, & \text{if } i \geq 1. \\ \frac{P_{m0}}{P_{m1} - P_{m0}} = \frac{\mu}{\lambda_h + \alpha - \mu}, & \text{if } i = 0. \end{cases} \quad (5)$$

식(5)에서

$$\frac{P_0}{P_{m1} - P_{m0}} \frac{P_{m1} - P_{m0}}{P_{m2} - P_{m1}} \dots \frac{P_{m(i-1)} - P_{m(i-2)}}{P_{mi} - P_{m(i-1)}} = \frac{\mu}{\lambda_h + \alpha - \mu} \left(\frac{\mu}{\lambda_h + \alpha} \right)^{i-1}$$

와 같이 쓸 수 있다.

정리하면

$$P_{mi} - P_{m(i-1)} = P_{m0} \left(\frac{\lambda_h + \alpha - \mu}{\mu} \right) \left(\frac{\lambda_h + \alpha}{\mu} \right)^{i-1}. \quad (6)$$

식(6)에서 일반항 P_{mk} 를 구하기 위하여 좌항과 우항의 i 를 1부터 k 까지 합하면 다음과 같은 식(7)이 전개되어 진다.

$$\begin{aligned} \sum_{i=1}^k [P_{mi} - P_{m(i-1)}] &= P_{mk} - P_{m0} \\ &= \frac{\frac{\lambda_h + \alpha - \mu}{\mu} P_{m0} \left(1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \right)}{1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)} \end{aligned} \quad (7)$$

식(7)을 정리하면

$$\begin{aligned} P_{mk} &= P_{m0} + \frac{\frac{\lambda_h + \alpha - \mu}{\mu} P_{m0} \left(1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \right)}{1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)} \\ &= P_{m0} - P_{m0} \left(1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \right) \\ &= P_{m0} \left(\frac{\lambda_h + \alpha}{\mu} \right)^k. \end{aligned} \quad (8)$$

식(8)과 같이 버퍼의 각 상태값의 일반식을 얻을 수가 있다. 여기에서 P_{m0} 의 값을 구하기 위하여, buffer의 각 상태의 총합은 1이 되는 것을 이용한다[14].

$$\begin{aligned} \sum_{k=1}^N P_{mk} &= P_{m0} + P_{m0} \left(\frac{\lambda_h + \alpha}{\mu} \right) + P_{m0} \left(\frac{\lambda_h + \alpha}{\mu} \right)^2 \dots \\ &= \frac{P_{m0} \left(1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)^N \right)}{1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)} = 1 \\ \therefore P_{m0} &= \frac{\left(1 - \frac{\lambda_h + \alpha}{\mu} \right)}{1 - \left(\frac{\lambda_h + \alpha}{\mu} \right)^N} \end{aligned} \quad (9)$$

지금까지는 주버퍼의 각 상태 확률을 구하였다. 마찬가지로 제어버퍼의 상태확률은 주버퍼의 서비스율 μ 대신에 $\mu \cdot P_{m0}$ 를 대입하고 입력은 λ_i 만 있게 된다. 즉,

$$P_{c0} \lambda_i = P_{c0} P_{m0} \mu \quad (10)$$

$$\lambda_i P_{c0} + P_{c2} P_{m0} \mu = \lambda_i P_{c1} + P_{c1} P_{m0} \mu \quad (11)$$

$$P_{c1} = \frac{\lambda_i}{P_{m0} \mu} P_{c0} \quad (12)$$

그리고 주버퍼와 같은 방법으로

$$\begin{cases} \frac{P_{c0}}{P_{c1} - P_{c0}} = \frac{P_{m0} \mu}{\lambda_i - P_{m0} \mu}, & \text{if } k = 0. \\ \frac{P_{ck} - P_{c(k-1)}}{P_{c(k+1)} - P_{ck}} = \frac{P_{m0} \mu}{\lambda_i}, & \text{if } k = 0. \end{cases} \quad (13)$$

와같이 전개할 수 있다.

위 식(13)에서

$$P_{ck} = P_{c0} \cdot \left(\frac{\lambda_l}{P_{m0} \mu} \right)^k \quad (14)$$

$$P_{c0} = \frac{\mu \cdot P_{m0} - \lambda_l}{\mu \cdot P_{m0} - 1 - \left(\frac{\lambda_l}{\mu \cdot P_{m0}} \right)^L} \quad (15)$$

$$P_{ck} = \frac{\mu P_{m0} - \lambda_l}{\mu P_{m0} - 1 - \left(\frac{\lambda_l}{\mu P_{m0}} \right)^L} \left(\frac{\lambda_l}{\mu P_{m0}} \right)^k \quad (16)$$

과 같이 식이 전개되어진다. 식(8)과 식(16)의 값을 구하기 위하여 버퍼의 상태방정식의 전개식, 즉 P_{mk} , P_{ck} 를 μ 는 1로 $\lambda_h, \lambda_l, N, L$ 의 값들은 상수로 가정하고 α 의 식으로 전개를 하면

$$P_{mk} = \frac{\mu - (\lambda_h + \alpha)}{\mu - (\lambda_h + \alpha)^N} \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \quad (17)$$

$$P_{ck} = \frac{1 - \frac{\mu - (\lambda_h + \alpha)^N}{\mu - (\lambda_h + \alpha)} \lambda_l}{1 - \left[\frac{\mu - (\lambda_h + \alpha)^N}{\mu - (\lambda_h + \alpha)} \lambda_l \right]^L} \left[\frac{\mu - (\lambda_h + \alpha)^N}{\mu - (\lambda_h + \alpha)} \lambda_l \right]^L \quad (18)$$

결과적으로, 각 서비스에서 영향을 미치는 지연값은 각 버퍼의 처리 시간의 값과 같으므로 식(19)과 (20)에 대입하여 각 서비스별 지연값을 구할 수 있다.

$$\begin{aligned} Delay_h = Delay_m &= \sum_{k=0}^N k \cdot P_{mk} \\ &= \sum_{k=0}^N k \cdot \frac{\mu - (\lambda_h + \alpha)}{\mu - (\lambda_h + \alpha)^N} \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \end{aligned} \quad (19)$$

$$\begin{aligned} Delay_l &= Delay_m + Delay_c \\ &= \sum_{k=0}^N k \cdot P_{mk} + \sum_{k=0}^L k \cdot P_{ck} \\ &= \sum_{k=0}^N k \cdot \frac{\mu - (\lambda_h + \alpha)}{\mu - (\lambda_h + \alpha)^N} \cdot \left(\frac{\lambda_h + \alpha}{\mu} \right)^k \\ &\quad + \sum_{k=0}^L k \cdot \frac{1 - \frac{\mu^N - (\lambda_h + \alpha)^N}{\mu^N \cdot [\mu - (\lambda_h + \alpha)]} \cdot \lambda_l}{1 - \left[\frac{\mu^N - (\lambda_h + \alpha)^N}{\mu^N \cdot [\mu - (\lambda_h + \alpha)]} \cdot \lambda_l \right]^L} \cdot \left[\frac{\mu - (\lambda_h + \alpha)^N}{\mu - (\lambda_h + \alpha)} \cdot \lambda_l \right]^k \end{aligned} \quad (20)$$

식(19) 및 (20)의 결과에 따른 지연값은 그림 3, 4, 5

와 같다. 그림 3은 실시간성과 비실시간성 트래픽의 비율이 1:1인 경우의 그래프로서 H_K(5)는 L의 값이 5일 경우의 실시간성 트래픽의 지연값을 의미한다. 그리고 L_K(5)는 L의 값이 5일 경우의 비실시간성 트래픽의 지연값을 나타내고 있다.

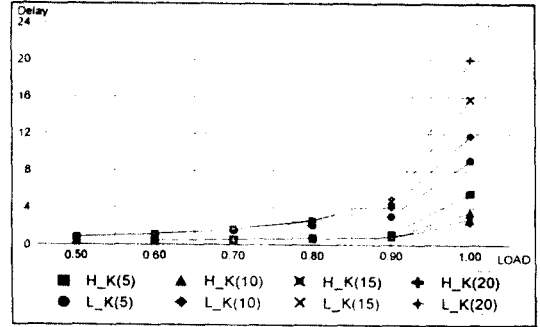


그림 3. 수식에 의한 HOL-LJ의 지연값(H:L=1:1).
Fig. 3 Delay of HOL-LJ using equation(H:L=1:1)

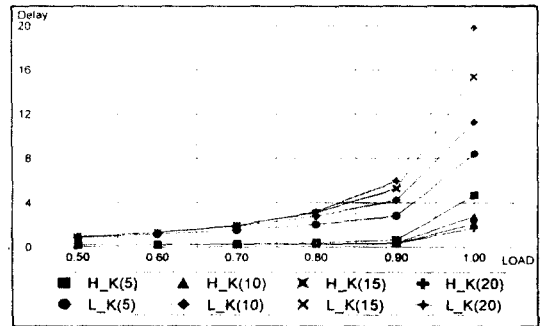


그림 4. 수식에 의한 HOL-LJ의 지연값(H:L=1:3).
Fig. 4 Delay of HOL-LJ using equation(H:L=1:3)

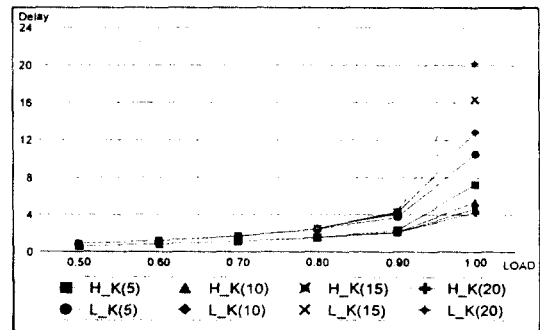


그림 5. 수식에 의한 HOL-LJ의 지연값(H:L=3:1).
Fig. 5 Delay of HOL-LJ using equation(H:L=3:1)

3. HOL-LJ의 하드웨어 구성

기존에 제안되어 온 알고리즘과 비교하여 볼 때 HOL-LJ방법은 하드웨어 구성면에서 상당히 간단하여 고속처리가 기본인 ATM망에서 효율적으로 사용할 수 있다. $L=X$ 인 경우, 제안 방법에서 필요로 하는 하드웨어는 큐(queue)의 기억장소와 X 개의 비실시간 트래픽에 대한 정보를 처리할 수 있는 제어장치만 있으면 구현 가능하다. 하드웨어 구성도는 그림 6과 같다. 동작 방법은 새로운 입력이 들어오면, 이 셀이 시간우선순위에서 High인지 아니면 Low인지를 판별하여, High인 경우 바로 큐에 입력이 되고, Low이면 비실시간 트래픽을 처리하는 큐로 입력이 된다. 만일 비실시간 큐가 가득차 있을 경우 가장 앞의 셀을 큐로 입력시키고 새로운 입력을 받아들인다. 서비스 측면에서는 큐에 전송할 셀이 존재하면 큐에서 전송하고, 그렇지 않으면 비실시간 큐에서 전송을 한다.

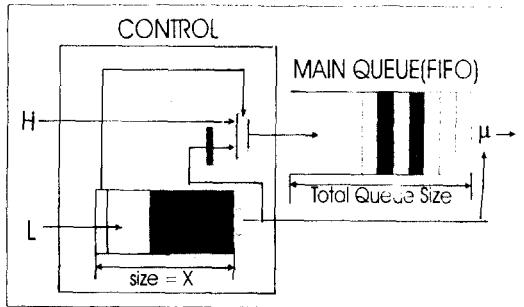


그림 6. HOL-LJ의 하드웨어 구성도
Fig. 6 Hardware component of HOL-LJ

III. 시뮬레이션을 통한 성능분석

점프 계수 L 값의 변동은 버퍼에서의 우선순위 고려 정도와 구현 복잡도가 상관적인 관계를 가지고 있으므로 각 서비스의 QoS를 만족하는 최소의 L 값의 설정이 필요하다. 예를 들어 $L=X$ 인 경우 마지막 실시간 트래픽 셀의 위치와 X 개의 비실시간 트래픽 셀 위치만 알고 있으면 HOL-LJ 방법을 실현할 수 있다. 여기에서 X 값이 클수록 실시간 트래픽의 셀을 더욱 빠르게 서비스를 해줄 수 있으나, 저장해야 할 비실시간성 트래픽의 개수가 증가하기 때문에 하드웨어 구현 면에서는 부담이 된다.

L 값에 따른 각 트래픽의 지연 형태를 알아보기 위해 시뮬레이션을 실행하였다. 실시간성 트래픽과 비실시간성 트래픽의 발생형태는 각각 포아송분포로 가정하였으며, 시간우선에서의 부하비율은 실시간성 트래픽과 비실시간성 트래픽이 동일한 경우와 1:3, 3:1인 경우를 가정하였다. 그리고 기존에 제안된 시간우선순위의 제어 방법과 비교하기 위하여 같은 환경 하에서 HOL과 HOL-PJ를 수행하였다.

먼저 실시간성 트래픽과 비실시간성 트래픽의 비율이 1:1인 경우를 살펴보면, 그림 7에서 제안하는 방법과 HOL, HOL-PJ 방법의 성능을 비교하고 있다. H-LJ(7)은 L 값이 7인 실시간성 트래픽의 지연값을 나타내고, L-LJ(7)은 L 값이 7인 비실시간성 트래픽의 지연값을 나타내고 있다. 그리고 H-HOL은 HOL의 실시간성 트래픽을 의미하고 L-PJ(7)은 HOL-PJ에서 임계값이 7인 비실시간성 트래픽의 지연값을 나타내고 있다. 결과치를 살펴보면, LJ의 경우 실시간성 트래픽이 L 값에 따라 지연값이 9셀 지연에서 점점 더 감소됨을 알 수 있고, 비실시간성 트래픽의 지연값은 증가됨을 알 수 있다. 비교대상으로 임계값이 7인 HOL-PJ의 실시간성 트래픽의 지연값은 12셀 지연을 보이고 있다. 이는 HOL-PJ가 복잡한 하드웨어를 가지고도 성능이 HOL-LJ 보다 저하됨을 살펴볼 수 있다.

부하율이 실시간성 트래픽과 비실시간성 트래픽의 비율이 1:3인 경우(그림 8), 먼저 HOL-LJ에서는 실시간성 트래픽은 부하율이 1:1인 경우와 거의 변동이 없고, 비실시간 트래픽의 지연 변동이 L 값에 의하여 적은 변동(16~20셀 간격)을 보이고 있다. 그리고 HOL과 HOL-PJ에서는 임계값이 증가함에 따라 비실

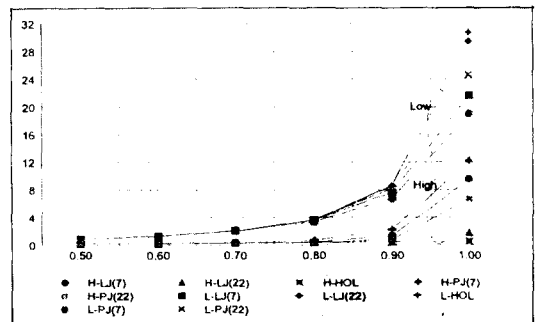


그림 7. 부하비율이 R:N=1:1인 경우
Fig. 7 Delay in case of R:N=1:1

시간성 서비스는 완만한 증가를 보이고 있으나 실시간 서비스의 경우는 임계값의 값이 4씩 변화함에 따라 지연이 2셀 간격씩 감소함을 알 수 있다. 결과 그림에서도 알 수 있듯이 실시간 서비스의 비율이 적고 비실시간 서비스가 많을 때에는 경계값(HOL-LJ에서는 L값, HOL-PJ에서는 임계값)에 의하여 비실시간 서비스의 변동이 적음을 파악할 수 있다. 그리고 비실시간 서비스의 지연시간은 부하비율이 동일할 경우와 비교하여 부하가 1.0일 때에 약 8셀 간격이 감소함을 알 수 있다.

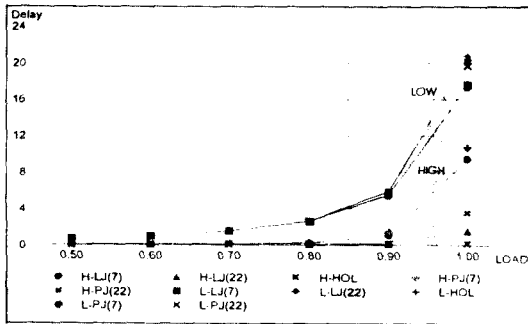


그림 8. 부하비율이 R:N=1:3인 경우
Fig. 8 Delay in case of R:N=1:3

부하비율이 실시간 서비스가 비실시간 서비스보다 3배 많을 경우(그림 9)에는 같은 부하율에서와 같은 지연값을 보이고, 비실시간 서비스의 지연값이 상당히 늘어남을 알 수 있다. 또한 HOL의 방법에서는 비실시간 서비스의 지연값이 부하가 1.0에서 60셀이나 되어 상당한 서비스 품질의 저하를 가져옴을 알 수 있다.

전체적으로 살펴보면, L값이 커지면 HOL의 성능으로 근접해지며, 따라서 비실시간 트래픽에 대한 고려가 전혀 안되어 지연값이 상당히 증가함을 알 수 있다. 또한 HOL-PJ의 임계값과 제안한 방법의 L값이 같은 경우를 살펴보면 제안한 방법이 실시간 서비스의 특성을 더욱 부각시키는 장점을 가지고 있음을 알았다. 이로서 HOL-LJ의 간단한 하드웨어를 가지고 HOL-PJ 보다 나은 성능을 가지게 할 수 있음을 알 수 있다.

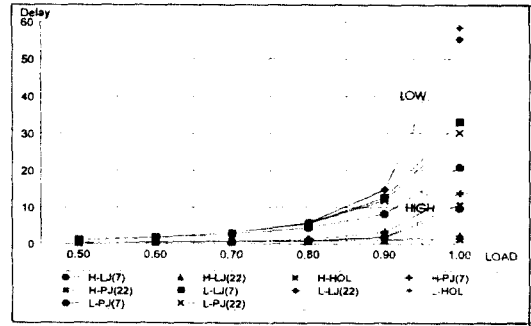


그림 9. 부하비율이 R:N=3:1인 경우
Fig. 9 Delay in case of R:N=3:1

IV. HOL-LJ와 손실우선순위와 연동

최근 연구동향은 두 가지 우선순위, 즉 시간 우선순위와 손실 우선순위를 동시에 수행하는 방안들이 제안되어 오고 있다. 이는 각 채널의 트래픽 특성을 바탕으로 서비스를 제공하기 위한 것으로서, 각 채널이 요구하는 서비스 품질(QoS)에 대해 만족할 만한 서비스를 제공하기 위해서는 두 가지 우선순위에 대한 고려가 필수적이다. 따라서 본 논문에서는 제안한 시간우선순위 제어기법을 기존의 손실우선순위 제어기법에 적용하여 위의 기능을 수행하는 버퍼 제어 기법을 제안한다.

1. HOL-LJ와 이중버퍼방법과 혼합

HOL-LJ 방법과 임계치를 가지는 이중 버퍼를 사용하면, 시간우선순위 제어기능과 손실 우선 제어기능을 수행할 수 있다. 먼저 설명의 편의를 위하여 실시간 손실우선순위를 RH로, 실시간 손실저우선순위를 RL로, 비실시간 손실우선순위를 NH, 마지막 비실시간 손실저우선순위를 NL로 정의한다. 기존의 방법에서는 시간우선순위 제어에 의해 손실제어의 효과를 함께 노렸으나, 이 경우 손실제어의 큰 이득을 얻을 수는 없다. 즉 실시간 트래픽의 서비스의 손실 감소는 RL 트래픽의 손실 역시 줄어들음을 의미하므로 손실 제어의 효과는 크다고 할 수 없다. 또한 각 버퍼에서 따로 처리되기 때문에 다른 쪽 버퍼에 우선순위가 높은 셀이 별로 없더라도 낮은 우선순위 셀의 손실률이 커지지만 고려된 방법에 따르면 낮은 우선순위 셀들의 손실률은 언제나 비슷하게

유지된다.

본 방법에서 손실우선순위제어는 임계치에 의한 서비스율의 변동에 의해 수행되며 큐 길이에 의해서 제어되기 때문에 HOL-LJ 알고리즘과 마찬가지로 하드웨어의 간편성을 가지게 된다. 이는 고속처리를 기본으로 하는 ATM 방식에서 필수적인 사항이다. 3개의 임계치를 기준으로 버퍼 서비스율이 결정되며 (RH + NH)버퍼와 (RL + NL)버퍼의 전송비율 x:y는 큐 길이에 의해 조정되며 다음과 같다.

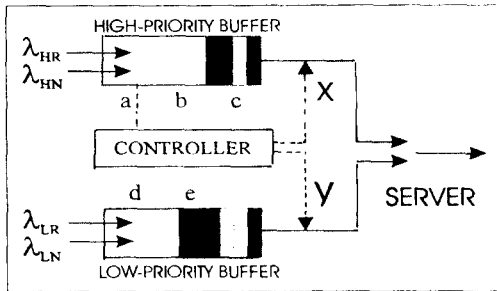


그림 10. HOL-LJ를 사용한 버퍼제어기법
Fig. 10 Merging HOL-LJ with double buffer

1. c, e ==> 1:1
2. c, d ==> 1:n
3. b, e ==> n:1
4. b, d ==> 1:1
5. a, e ==> n:0
6. a, d ==> n:1

그림 10은 전체적인 제어형태를 보여준다. 그러나 이러한 제어구조는 이중버퍼를 제어하므로 단일 버퍼를 사용하는 방법보다 제어기의 구조를 복잡화시키는 단점이 있다.

제안된 알고리즘에 따라 네 가지 형태의 트래픽 (RH, RL, NH, NL)에 대한 셀 지연 및 셀손실율의 형태를 고려하였다. JUMP 계수 L=11, 버퍼 크기는 각각 16셀 크기를 사용하였으며, 전송비는 3:1로 고정하였다. 그리고 전체 부하율을 변화시키면서 시뮬레이션을 실행하였다(그림 11, 그림 12).

결과를 살펴보면, 실시간성 트래픽은 적은 지연을 가지며, 손실 우선순위에 따라 각각 손실률을 가짐을 보여준다. 즉 지연면에서 살펴보면, RH와 RL가 부하가 커짐에도 4셀 간격 이내의 지연값을 가짐을 알 수 있고 상대적으로 NH는 부하가 1.0이 될 때 급격히 증

가하여 12셀 간격의 지연값을 가지게 된다. 그리고 NL역시 상당히 많은 지연값을 가지고 있다. 그리고 손실면에서 분석하면, 실시간성과 비실시간성 트래픽과는 무관하게 손실우선순위에 따라 균등한 손실률을 가지고 있음을 알 수 있다.

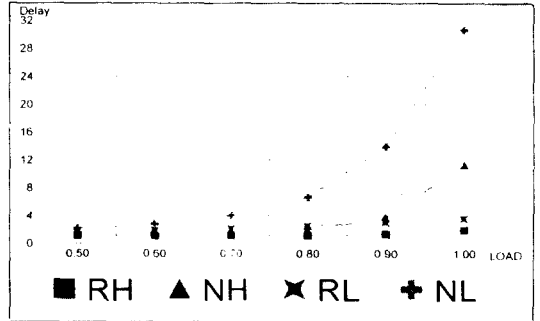


그림 11. HOL-LJ + 이중버퍼의 지연값
Fig. 11 Delay of HOL-LJ with double buffer

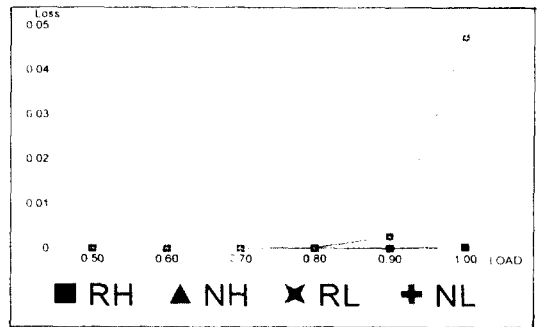


그림 12. HOL-LJ + 이중버퍼의 손실률
Fig. 12 Loss of HOL-LJ with double buffer

비교의 대상으로 QLT의 시뮬레이션을 수행하였다. QLT에서 큐의 길이는 실시간성 큐의 길이를 12로 비실시간성 큐의 길이를 20으로 정하고, 부하율 및 임계값을 변경해 가면서 시뮬레이션을 수행하였다. 결과는(그림 13, 그림 14)에 나타나 있다.

임계값이 14일 때의 결과를 살펴보면, 부하가 1.0일 때의 지연값은 실시간성 서비스는 손실우선순위에 관계없이 대략 5셀의 지연값을 가지게 되고, 비실시간성 서비스는 19셀 간격의 지연값을 가지고 있다. 그리고 임계값의 값이 커짐에 따라 지연값은 실시간

성 서비스에 대하여 더 적은 지연값을 가지도록 보장하고 있음을 알 수 있다. 손실면에서는 지연값의 결과와는 반대로 지연우선순위에서 낮은 우선순위의 셀이 적은 손실률을 가지고 있음을 알 수 있다. 그리고 손실에서는 임계값이 커짐에 따라 비실시간성 서비스의 손실률이 많아짐을 알 수 있다. 이는 실시간 큐에 우선순위를 가중시키기 때문이다. QLT 방법은 실시간성이면서 손실 우선순위가 높은 셀도 높은 손실률을 가지고 있는데, 이는 실시간성 버퍼와 비실시간성 버퍼의 구분에 의해 발생할 수 있는 손실제어의 약점을 나타내고 있다.

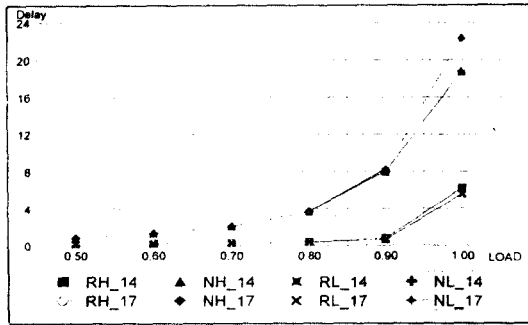


그림 13. QLT의 지연값
Fig. 13 Delay of QLT

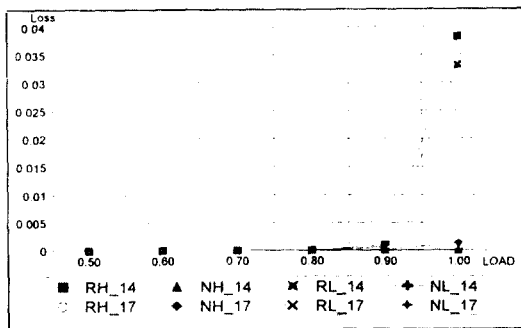


그림 14. QLT의 손실률
Fig. 14 Loss of QLT

2. PBS와 혼합 방안

손실 우선 순위를 위한 방법으로는 부분공유기법이 가장 많이 사용되고 있다. 이 방안의 주된 이점은 요구하는 품질을 만족하면서 하드웨어 구현이 매우

용이하다는데 있다[12][13]. 이점은 본 논문을 통해 제안하고 있는 HOL-LJ 방법도 같은 특성으로서 두 가지 알고리즘 모두 하드웨어 구현이 용이하기 때문에 각각의 우선순위제어기법을 통합한 제어 방법을 제안한다.

이를 통해 두 우선순위에 대한 제어기능을 가질 뿐만 아니라 고속 처리의 기능을 가질 ATM 다중화기의 하드웨어를 단순화시키게 된다. PBS를 구현하기 위해서는 버퍼의 채워진 정도를 나타내는 카운터만 가지면 되기 때문에 실제 이 두 가지 방법의 혼합방법도 매우 간단하게 구현할 수 있다.

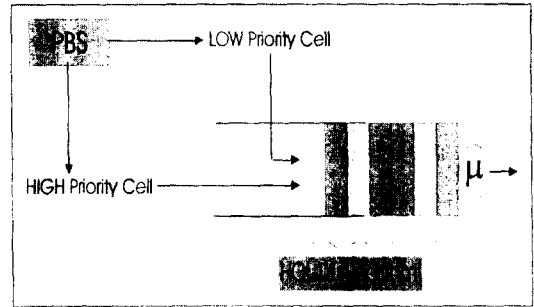


그림 15. PBS와 HOL-LJ의 혼합
Fig. 15 Merging HOL-LJ with PBS

PBS를 이용하는 HOL-LJ의 시뮬레이션을 수행하기 위하여 큐의 크기는 32셀로, L값은 일정 값으로 고정(본 논문에서는 14, 18, 22값으로 고정)하였다. 그리고 입력 트래픽 모델은 각 서비스 등급마다 독립적인 서비스로 인식하고 입력 분포는 포아송 분포를 가지고 도착한다고 가정하였다.

결과를 살펴보면 PBS와 HOL-LJ의 혼합으로 인하여 셀의 손실률은 PBS의 손실률과 같은 값(그림 16)을 나타내고 있고, 셀 지연값(그림 17)은 HOL-LJ 방법에서의 지연값과 같은 값을 갖고 있다. 이는 PBS 방법은 손실우선순위를 제어하기 위한 방법이고 HOL-LJ는 지연우선순위를 위한 방법이기 때문이다. 그리하여 각 서비스등급별 서비스 품질을 고려해 보면 실시간 서비스는 적은 지연값을 가지고 서비스를 받도록 보장하고 있고, 손실우선순위가 높은 서비스는 적은 손실률로서 서비스를 받도록 하고 있음을 알 수 있다.

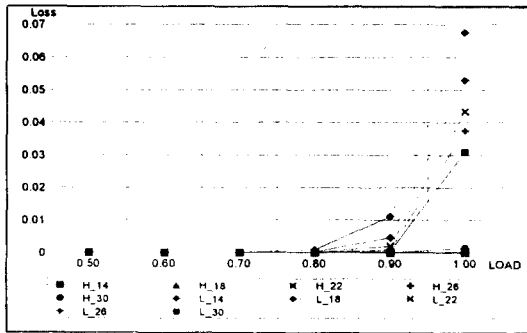


그림 16. PBS + HOL-LJ의 손실률
Fig. 16 Loss of HOL-LJ with PBS

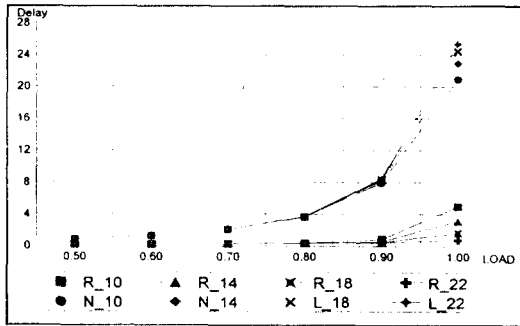


그림 17. PBS + HOL-LJ의 지연값
Fig. 17 Delay of HOL-LJ with PBS

V. 결론

ATM망에서는 다양한 서비스 트래픽을 통계적 다중화 방식으로 전송하므로 망 자원을 효율적으로 이용할 수 있지만 소요대역이나 서비스 요구품질이 각기 다른 다양한 흐름이 망자원을 공유하여야 하고, 고속의 셀을 전송함에 따라 망 차원의 폭주상태가 발생할 수 있고, 예측할 수 없는 트래픽의 변화로 인한 셀 전송지연 및 셀 손실도 발생할 수 있다. 따라서 사용자가 요구하는 서비스 품질을 충족시키고 망 자원을 효율적으로 사용하기 위해서는 효과적인 트래픽 제어가 필수적이다. 기존에는 손실우선순위를 중심으로 폭주발생시 그 상황을 벗어나기 위한 방법이 주로 연구되어 왔다. 그러나 최근에는 QoS class를 4가지 종류로 나누어 각각의 서비스 품질을 만족시키고자 하는 제어방법이 제안되고 있다.

본 논문에서는 기존에 제안된 시간우선순위 제어방법인 HOL-PJ를 개선하여 간단한 하드웨어 구성을 가지고 제어가 가능한 HOL-LJ 알고리즘을 제안하였다. 제안한 HOL-LJ의 성능을 시뮬레이션을 통하여 분석한 결과 실시간성 트래픽과 비실시간성 트래픽의 부하비율이 1:1인 경우는 25%정도, 부하비율이 1:3인 경우에는 15%, 3:1인 경우에는 30%의 성능이 우수함을 살펴볼 수 있었다. 즉 전체적으로 살펴볼 때 간단한 하드웨어 구성을 가지고도 HOL-PJ의 성능보다 25% 정도 더 우수함을 알 수 있었다(그림 3, 4, 5 참조). 또한 제안한 알고리즘은 단일 버퍼를 사용하는 방법임으로 기존에 제안된 손실우선순위의 제어방법과 혼합하여 사용함으로써 4종류 트래픽의 QoS를 효과적으로 보장할 수 있다. 그리고 제안한 HOL-LJ 알고리즘을 손실우선순위 제어기법에 적용시켰을 때, PBS와 혼합하는 방법이 이중버퍼와 혼합하는 방법보다 성능이 우수한 것으로 나타났다. 따라서 본 논문에서는 각 채널이 요구하는 지연과 손실면에서 서비스 품질을 만족시키기 위해, 단일버퍼에서 손실에 기준하여 분리하고 전체 버퍼에서 시간우선순위제어를 수행하는 기법을 제안하였다. 제안한 방법을 통해 적은 버퍼크기와 간단한 제어방법으로 시간우선순위 제어와 손실우선순위 제어를 동시에 할 수 있게 된다.

본 논문에서 제안한 HOL-LJ 알고리즘과 기존에 PBS 알고리즘을 혼합하여 버퍼의 제어부를 구성한 결과 다음과 같은 서비스 요구사항을 만족할 수 있었다.

- 실시간성 고우선순위 트래픽 → 적은 지연, 적은 손실
- 비실시간성 고우선순위 트래픽 → 적은 손실
- 실시간성 저우선순위 트래픽 → 적은 지연
- 비실시간성 저우선순위 트래픽 → 적은 손실
- 연결모드 채널 → 셀 순서 유지
- 간단한 제어 및 하드웨어 구현 간편성

향후의 연구과제로는 첫째, 입력되는 각 트래픽 등급의 호 발생빈도에 따라 능동적으로 점프계수(L)를 변경할 수 있는 방법과 둘째, 다양한 트래픽(포아송 이외) 발생 형태에 대한 분석에 관하여 연구가 계속 되어야 할 것이다.

참 고 문 헌

1. ITU-T Draft Recommendation I.371. "Traffic control and congestion control and in B-ISDN", SG X VIII, Geneva, Switzerland, June 1992.
2. S. E. Minzer, "Broadband ISDN and Asynchronous Transfer Mode (ATM)," IEEE Comm. magazine, pp.17-24, Sep. 1989.
3. Ryutaro Kawamura, Ken-ichi Sato, and Ikuo Tokizawa, "Self-Healing ATM Networks Based on Virtual Path Concept", IEEE Commun.Mag., Jan. 1994.
4. J.J. Bae and T.Suda, "Survey of Traffic Control Schemes and Protocols in ATM Networks", proceedings of the IEEE, vol. 79. no.2, pp.170-189, February, 1991.
5. ITU-T Draft Recommendation I.361. "B-ISDN ATM Layer Specification", SG X VIII, Geneva, May 1990.
6. ITU-T Draft Recommendation I.362. "B-ISDN Adaption Layer Functional Description", 1990.
7. Setia Yazid and H. T. Mouftah, "Congestion control Methods for B-ISDN", IEEE Comm. Mag., July. 1992.
8. C. Anthony Cooper and Kun I. Park, "Toward a Broadband Congestion Control Strategy". IEEE Network Mag. May, 1990.
9. H. Kroner, G. Hebuteme, P. Boyer, and A. Gravey "Priority Management in ATM Switching Nodes", IEEE JSAC, Vol. 9, No. 3, pp.418-427, April, 1991.
10. David W. Petr, James, and Wineinger "End-to-end Priority Cell Discarding Analysis for ATM Networks", INFOCOM, Vol. 3, pp.1178-1185, 1993.
11. Renu Chipalkatti, James F. Kurose, Don Towsley, "Scheduling Policies for Real-Time and Non-Real-Time Traffic in a Statistical Multiplexer", Proc. IEEE INFORM '89, pp. 774-783, 1989.
12. Felix Hartanto, Harsha R. Sirisena, Krzysztof Pawlikowski, "Protective Buffer Policies at ATM Switches", ICC, pp.960-964., 1995.
13. Hans Kr ner, "Comparative Pformance Study of Space Priority Mechanisms for ATM Networks", INFOCOM, pp.1136-1143, 1990.
14. L. Kleinrock, Queueing System I & II, 1976.
15. Hyun-Chul Cha and Ki-Jun Han, "HOL-T Priority Control Scheme for ATM switch", JCCI, Vol. 2, pp.197-202, 1992.
16. 박홍식, "ATM 트래픽 제어기술", 전자공학회지, 1993, 12.
17. Sang Gug Jung, Yong Ohk Chin, "Congestion Control with the Double & Hysteresis Threshold in ATM Networks", GLOBECOM, pp.595-599, 1994.
18. R.Beraldi and S.maranao. "Limiting Removal Depth in the Pushout Scheme for ATM Networks". In Proc. of IEEE ICC, pp.997-1003, Chicago, MI, June 1992.
19. 김현철, 이민구, 이문호, "ATM망에서 HOL-LJ알고리즘을 이용한 효율적 버퍼제어기법", 한국통신학회 하계종합학술대회, 1996.
20. HyonChol Kim, YoungIl Jeong, MyungRyoung Moon, JaeHwan Hong, MoonHo Lee, "An Efficient Buffer Control Algorithm in ATM Networks", ICCS/ISPACS, 1996.

金 賢 哲(HyonChol Kim)준회원

1996년 2월: 전북대학교 정보통신공학과 졸업(공학사)

1996년 3월~현재: 전북대학교 대학원 정보통신공학과 석사과정 재학중

※ 관심분야: ATM 네트워크, 무선네트워크

文 明 龍(MyungRyong Moon)정희원

1982년 2월:조선대학교 전기공학과 졸업(공학사)

1984년 2월:조선대학교 대학원 전기공학과 졸업(공학석사)

1996년 3월~현재:전북대학교 대학원 영상정보공학과 박사과정 재학중

현재:전주공전 전기공학과 조교수

※관심분야:초고주파 통신, 무선 네트워크

李 門 浩(Moon Ho Lee)정희원

1970년~1980년:남양문화방송(주) 송신소장

1985년 8월~1986년 8월:미국 미네소타주립대 전기과 포스터닥터

1995년 12월~1996년 12월:독일 하노버 아혼공대 전기과 연구교수

일본동경대학 정보통신공학박사, 통신기술사

현재:전북대 정보통신공학과 교수

※관심분야:디지털 이동통신, 영상통신, 무선 ATM 네트워크