

# VC-Merge Capable ATM Switch의 버퍼용량 축소에 관한 연구

정회원 유정욱\*, 조양현\*\*, 오영환\*\*\*

## A Study on Reducing Buffer for VC-Merge Capable ATM Switch

Jung-wook Yoo\*, Yang-hyun Cho\*\*, Young-hwan Oh\*\*\* *Regular Members*

### 요 약

레이어2 스위칭과 레이어3 라우팅의 통합 모델로써 MPLS(Multi-Protocol Sabel Switching) 환경에서 ATM LSR(Label-Switching Routers)은 백본망에서의 고속 전송이 가능하여 현재의 라우터 구조로써 제안되어지고 있다. MPLS가 코어 라우터로써 적용이 될 경우 확장성을 위해 label merging이라는 기술이 필요하다. VC(Virtual Circuit) merging은 ATM LSR에서 많은 IP 라우터를 하나의 라벨로 매핑을 시키며 수천 개의 목적지에 전송 할 수 있는 확장성 있는 매핑 기술이다. VC merging은 같은 목적지인 다른 패킷들 간의 셀들의 섞임을 방지하기 위해 재 조합 버퍼가 요구된다. 재 조합 버퍼 사용시 일시적인 체증 현상이 발생하며 Non-VC merging과 비교 시 많은 셀 손실과 많은 버퍼를 요구하게 된다.

본 논문에서는 RED(Random Early Detection)알고리즘을 적용하여 VC merging시 필요한 버퍼의 요구 량과 셀 손실을 줄였다.

### ABSTRACT

In a multiprotocol label switching(MPLS) domain which is the paradigm of integrating layer-3 routing with layer-2 switching, ATM LSR(Label-Switching Routers) has been proposed as a today's router architecture because of possibility of providing the highest forwarding capacity in the backbone network. When MPLS is applied to the design of large core routers, label merging is required in order to support scalability. Virtual Circuit(VC) merging is a mechanism in an ATM-LSR that allows many IP routers to be mapped to the same VC label and provides a scalable mapping method that can be transmitted thousands of destinations. VC merging requires reassembly buffer so that cells among different packets for the same destination do not interleave with each other. Using reassembly buffer is a direct result of temporary congestion and cell loss and buffer requirement of VC merging can be extended when compared to non-VC merging.

This paper using RED(Random Early Detection) algorithm reduced buffer requirement and cell loss in VC merging.

### 1. 서론

최근 정보화 사회로의 급속한 변천과 이를 뒷받침할 수 있는 기술적 배경 등이 상호 상승작용을 하면서 인터넷의 이용이 기하급수적으로 증가되고 있다. 망이 이러한 인터넷 이용 인구의 확산을 지원

하기 위해서 해결해야 할 주요 과제로는 고속 통신을 가능케 할 대역폭의 확대, 불특정 다수에게 전송하는 공간적 및 사용자 수의 증가를 수용하기 위한 망의 확장성, 그리고 다양한 서비스를 제공할 수 있는 기능 등을 들 수 있다.

MPLS의 등장배경은 지금까지 구현된 Classical

\* Eastelsystems (yoojw@eastelsystems.com)

\*\* 삼육대학교

\*\*\* 광운대학교

논문번호 : 00289-0726, 접수일자 : 2000년 7월 26일

IP over ATM, LANE(LAN Emulation), MPOA (Multi-protocol Over ATM)등에서 제기되었던 여러 가지 문제들인 망의 확장성, QoS의 보장, 트래픽 엔지니어링, 고속통신의 저 비용 실현 등이 있다.

MPLS는 개념적으로는 네트워크 계층이나 특정 데이터 링크 기술에 국한되는 것이 아니지만, 현재로는 ATM망이 QoS의 보장 및 트래픽 엔지니어링에 있어서 구현상의 장점을 갖고 있으므로 주요 적용 대상으로써 관심의 초점이 되고 있기 때문에 본 논문에서는 ATM 기반의 MPLS를 가정한다. MPLS의 주요 관심사중 망의 확장성은 인터넷 사용자의 증가로 인한 라우팅 노드 수의 증가에 따른 라우팅 트래픽의 증가, 라우팅 테이블 크기의 증가, 노드간의 연결 수 급증 등에 따른 망의 확장성을 해결할 수 있어야 한다. 그중 많은 수의 Edge router들간의 연결(connection)을 효율적으로 확장할 수 있는 방안을 제공하기 위해서 MPLS에서는 merging이라는 기술을 사용한다<sup>[1]</sup>.

하지만 label merging을 ATM-LSR(Label Switched Router) 영역에서 적용하는 데에는 문제가 있다. 왜냐하면 ATM-LSR 영역에서는 Ingress에서 Label된 패킷이 셀 단위로 나뉘어져 전송된다. 따라서, 여러 개의 다른 상위 노드들로부터의 트래픽들을 하나의 Label (VPI/VCI)로 Merging 하게 되면 다른 패킷으로부터의 셀들이 뒤섞이어서 (cell interleave) 차후에 패킷들을 재생할 수 없게 된다. 이러한 문제를 해결하기 위한 방안으로 다음과 같은 방법이 연구되어 왔다<sup>[2]~[4]</sup>.

VP merge는 여러 개의 입력 VP들을 하나의 VP로 Merge하고 다른 근원지로부터의 트래픽들에는 VP내의 다른 VCI값들을 할당하여 구별할 수 있게 하는 것이다. 이 방안을 사용하기 위해서는 각 VP내에 VCI공간의 조정이 필요하며 망의 확대에 따른 label의 부족현상도 일어 날 수 있다.

VC merge는 하나의 같은 VPI/VCI값으로 여러 개의 입력 Label(즉, VPI/VCI)을 Merge하는 방안이다. 그러나, 이 경우에는 다른 인터페이스로부터의 입력되는 다른 패킷에 속하는 셀들이 섞이는, 소위 Cell interleave가 발생하여 수신 노드에서는 각 패킷에 속하는 셀들을 구별해 낼 방법이 없게 된다. 셀들의 뒤섞임을 피하기 위해서는 Merge를 하는 노드에서는 하나의 패킷에 속하는 전체 셀들을 버퍼에 모은 후 항상 인접하게 연속적으로 전송함으로써 결과적으로는 패킷의 뒤섞임이 해소되게 한다. 이 경우, 수신 노드는 순서대로 수신하면 되고 다른

패킷들간의 경계는 AAL5의 end of CPCS PDU indicator에 의해 찾을 수 있다. 하지만 버퍼에 하나의 패킷이 다 모일 때까지 셀들을 모아야 되므로 버퍼용량이 증가하고 셀지연이 증가하게 된다<sup>[5]</sup>.

이런 문제점을 해결하기 위해서 본 논문에서는 현재 인터넷망에서 혼잡 회피 방법으로 쓰이고 있는 RED 알고리즘을 변형한 변형된 RED 알고리즘을 제안하였다. 변형된 RED 알고리즘 적용시 Output module에 control unit이 추가되며 RED 알고리즘에 의한 POS 셀 폐기 및 폐기된 셀에 대한 정보를 저장하여 이후 도착하는 해당 셀 폐기를 수행하게 된다.

## II. 관계이론

### 1. 스위치 구조

스위치에서 버퍼의 위치에 따라 Output buffer 방식과 Input buffer 방식으로 나눌 수 있는데, Input buffer 방식일 경우 패킷의 길이가 서로 다르므로 심한 HOL(Head Of Line) blocking 현상이 일어날 수 있다.

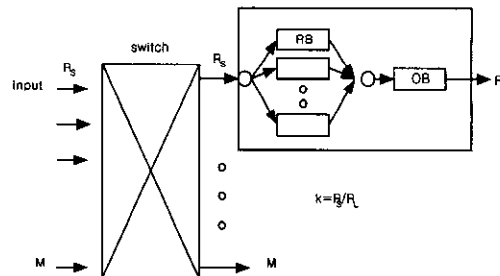


그림 1. 스위치 구조

따라서 본 논문에서의 스위치 구조로는 그림1과 같이 Output buffer 방식을 제안했으며, Merging 방식으로는 VC merging을 적용하였다.

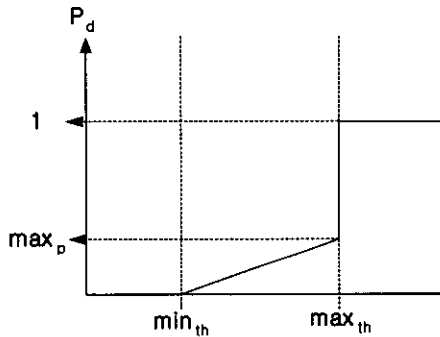
### 2. RED 알고리즘

RED는 버퍼에서의 평균 큐 길이(이후  $Q_{avg}$ 로 표기)를 이용하여 체중 제어를 수행한다. RED는 매 패킷이 도착할 때마다  $Q_{avg}$ 를 구하고 이것을 미리 정해놓은 파라미터인 최소 큐 한계값(minimum threshold, 이후  $min_{th}$ 로 표기)과 최대 큐 한계값(maximum threshold, 이후  $max_{th}$ 로 표기)과 비교한다.  $Q_{avg}$ 가  $min_{th}$  보다 작을 때에는 망의 링크의 사용이 낮은 수준에 머물러 있다고 판단할 수 있으

므로 모든 패킷은 정상적으로 처리된다. 하지만  $Q_{avg}$ 가  $min_{th}$ 와  $max_{th}$ 사이 에 있을 때 도착하는 패킷은 확률,  $p_a$ 에 의해 랜덤하게 폐기되든지 체증 통보(congestion notification)를 위한 비트(ECN bit)를 마크한다. 이 확률은 0에서부터 RED내에 정의된 상수인 최대 확률(maximum probability, 이하  $max_p$ 로 표기)까지의 범위 내에 존재한다.  $Q_{avg}$ 가  $max_{th}$ 보다 클 때에는  $Q_{avg}$ 가  $max_{th}$  밑으로 떨어질 때까지 도착하는 모든 패킷을 폐기하거나 ECN 비트를 마크한다.

$Q_{avg}$ 는 EWMA(Exponential Weighted Moving Average)를 통한 low-pass filter를 사용하여 다음과 같이 계산한다.

$$Q_{avg} \leftarrow (1 - w_q) \times Q_{avg} + w_q \times q$$



$$p = 0 \quad \text{if } k < \min_{th}$$

$$p = 1 \quad \text{if } k \geq \max_{th}$$

$$p = \frac{q - \min_{th}}{\max_{th} - \min_{th}} \times \max_p \quad \text{otherwise}$$

이와 같이 RED 알고리즘은 현재 인터넷망에서의 갑작스런 Bursty traffic에 대해 민감한 반응을 막고 또한 Global synchronization을 막기 위한 방법으로 연구되었다.

### III. VC merging 버퍼 해석

VC merging을 할 경우 그림 2와 같이 패킷이 형성 될 때까지 대기하고 있는 재 조합 버퍼와 Merging 후 보관 될 출력 버퍼로 구성되어 있다.

다음의 그림 3과 같은 시뮬레이션 모델을 구성하여 지연시간과 버퍼의 요구량을 비교해 보았다.

그림 3에서의 k값은 그림 1에서 정의된 것과 같이 각각의 포트의 입력라인( $R_s$ )과 출력라인( $R_t$ )의

Structure of Output Module

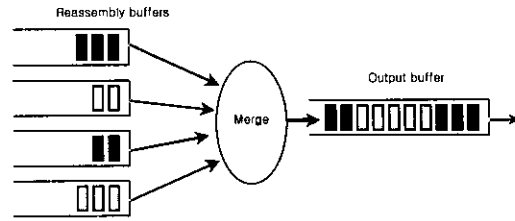


그림 2. 출력 모듈

비로써 나타내며 항상 1보다 작은 값으로 설정한다. 셀 지연 시간은 패킷의 첫 번째 입력 셀에서 부터 출력 버퍼에서 그 패킷의 EOP(End of packet) 셀 이 나갈 때까지의 시간이다.

$$D_{total}(total\ delay) = D_{RB}(delay\ in\ reassembly\ buffer) + D_{OB}(delay\ in\ output\ buffer)$$

분석 결과 링크효율이 0.6이하일 경우에 지연시간 이 VC merging을 할 경우 Non-VC merging인 경우와 많은 차이가 난다는 것을 알 수 있다<sup>[1]</sup>.

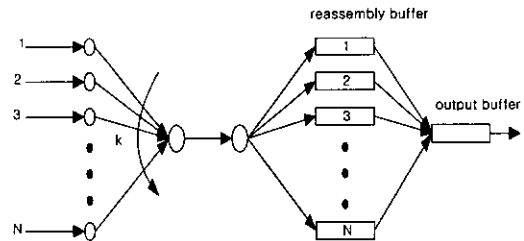


그림 3. 시뮬레이션 모델

### IV. 변형된 RED 알고리즘

본 논문에서는 IP 망에서의 혼잡 제어를 위한 Active queue management 알고리즘의 하나인 RED 알고리즘을 VC merging capable ATM switch에 맞게 변형시켰다.

RED 알고리즘을 VC merging capable ATM switch에 적용을 할 경우 폐기된 셀들에 대한 정보를 저장하고, 이후에 해당 입력 셀들을 폐기해야하는 Control unit이 Output module에 포함되어야 한다. 또한 Control unit에서 POS 셀 입력 시 패킷단위로 Average queue size를 계산하여 폐기 확률에 의해 셀 단위로 폐기시키게 된다. 셀 단위로 폐기됨에 따라 그 후에 도착하는 셀들을 폐기하여야 한다.

셀이 폐기되기까지의 과정은 다음과 같다.

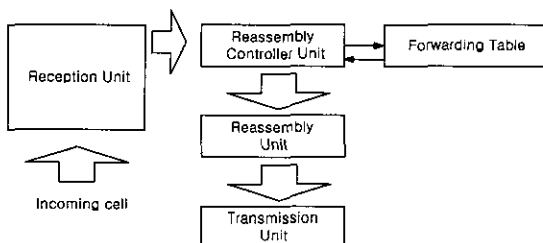


그림 4. 출력 모듈

그림 4에서 Output module은 Reception unit와 Reassembly control unit, Forwarding table, Reassembly unit, Transmission unit으로 구성되어 있다.

그림 5에서 보는 바와 같이, Output module에 RED 알고리즘을 적용한 경우의 모델이다. VC merging을 할 경우 재 조합 버퍼에서의 일시적인 체증 현상이 발생하며 Non-VC merging과 비교 시 많은 셀 손실과

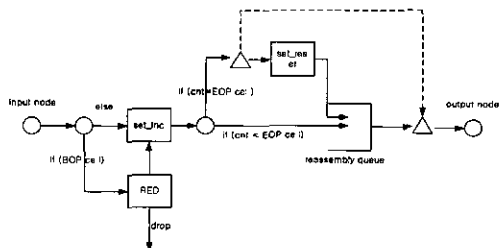


그림 5. RED 알고리즘을 적용한 Output module의 모델

많은 버퍼를 요구하게 됨을 알 수 있다. 이러한 이유로 RED 알고리즘을 적용하였고 merging 시스템에 RED 알고리즘을 각각의 셀 단위로 적용 할 경우 폐기되는 각각의 셀 정보를 알고 해당 셀(이전에 입력되어 버퍼에 있는 셀이나 이후에 이력 될 셀)을 폐기하도록 해야한다. 이런 경우 버퍼의 비효율성 및 제어하는데 있어 복잡성을 가져올 수 있으므로 패킷의 BOP 셀이 입력될 때만(패킷 단위로) RED 알고리즘을 적용하여 이후의 해당 셀을 모두 폐기시킨다.

VC merge capable ATM switch에 변형된 RED 알고리즘을 적용하면 아래와 같다.

*for new cell arrive*  
*look up the drop table*

*if dropped cell*  
*discard this cell*  
*else if*  
*accept this cell*  
*calculate average queue size*  
*if*  $\min_{th} \leq avg < \max_{th}$   
*calculate*  $P_o$   
*with*  $P_o$   
*mark/drop the arriving cell*  
*else if*  $\max_{th} < avg$   
*drop the arriving cell*

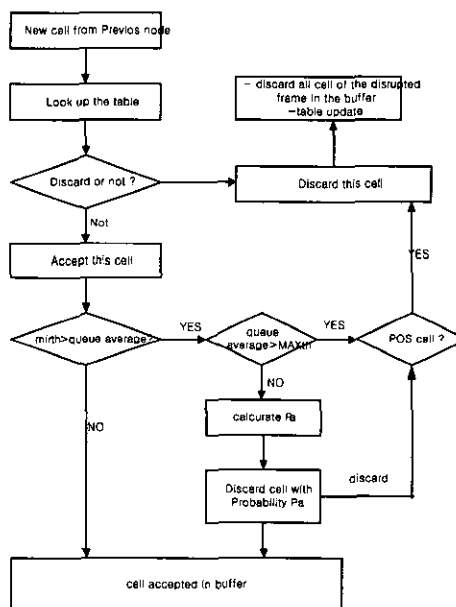


그림 6. 순서도

## V. 시뮬레이션

### 1. 시뮬레이션 환경

버퍼 용량을 18 packets 크기로 하였으며 최소 한계 값( $\min_{th}$ )을 5 packets, 최대 한계 값( $\max_{th}$ )을 10 packets로 설정하였다.  $w_q$  값은 0.002,  $\max_p$ 는 0.02로 설정하였다.

그림 7과 같은 네트워크 환경을 이용하여 시뮬레이션을 실시하였으며 EPD 알고리즘을 사용한 Non-VC merge capable switch(pure ATM switch), VC merge capable switch, RED 알고리즘을 이용

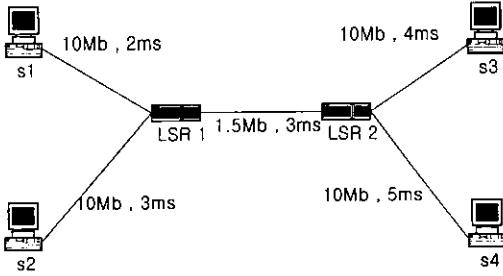


그림 7. 시뮬레이션 구성

한 VC merge capable switch를 Active queue length와 셀 지연에 대한 성능을 비교하였다.

### 1.1 Queue length

아래 그림 8, 9, 10과 같이 Bottleneck에서의 Non-VC merge capable ATM switch(EPD)와 VC merge capable ATM switch, RED 알고리즘을 적용한 VC merge capable ATM switch 각각의 Active queue length이며 그래프 중간의 가는 실선은 큐의 cumulative average를 나타낸다.

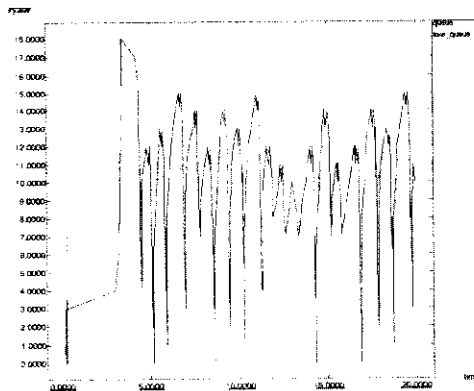


그림 8. VC merge capable ATM switch

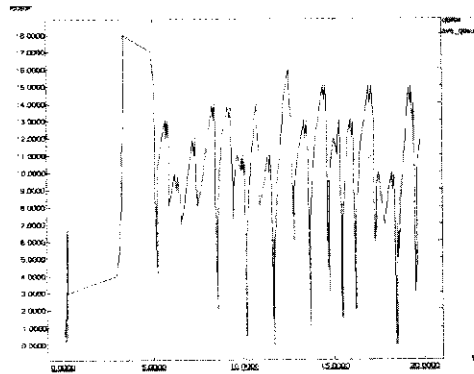


그림 9. EPD 알고리즘을 적용한 Non-VC merge capable ATM switch

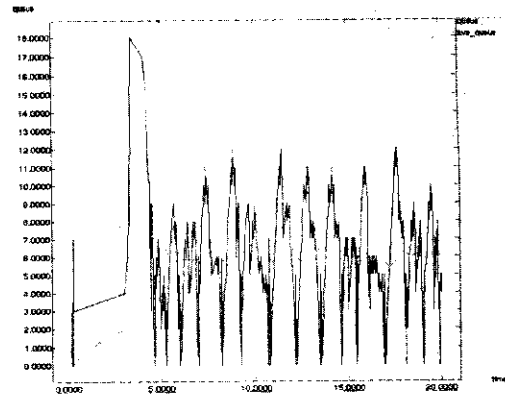


그림 10. RED 알고리즘을 적용한 VCmerge capable ATM switch

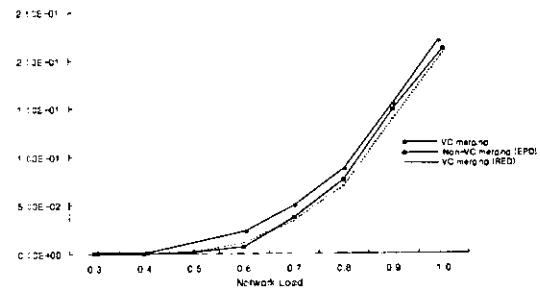


그림 11. 평균 패킷 손실율

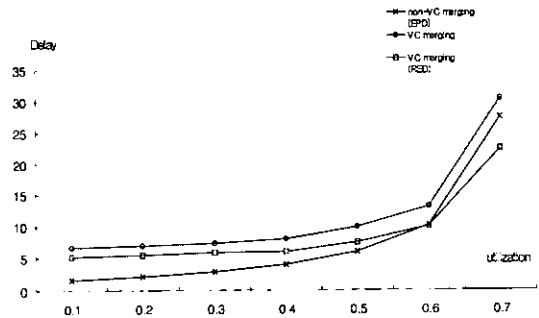


그림 12. 이용률에 따른 지연

같은 환경에서의 실험으로 인해 초기 값은 7 packets 정도로 같지만 이후의 결과 값에서 VC merge capable ATM switch의 요구되는 버퍼 량(그림 8)이 15 packets 정도이며 평균 queue 길이는 12 packets 정도로 Non-VC merge capable ATM switch(그림 9) 보다 많은 버퍼 량을 요구한다. 하지만 RED 알고리즘 적용 시(그림 10) 요구되는 버퍼 량은 11 packets 정도로 EPD 알고리즘의 경우 보다 작다는 것을 알 수 있다. 이와 같은 결과는 재

조합 버퍼에서 각각의 셀을 하나의 패킷으로 재 조합하는 과정에서 일시적으로 발생 할 수 있는 체증 현상에 인한 버퍼 량의 증가이다. 이러한 경우 RED 알고리즘에 의해 재 조합 시 발생하는 체증을 해소하며 약간의 요구되어지는 버퍼 량을 감소시킬 수 있음을 알 수 있다.

그림 11는 버퍼용량을 10, 18, 30 packets 3개의 파라미터로 정할 경우 네트워크 부하에 따른 평균 패킷 손실율에 대한 그래프이다.  $\rho=0.7$  이하일 경우 Non-VC merge capable ATM switch와 비슷한 결과를 보이다가  $\rho=0.7$ 이상일 경우 RED 알고리즘을 적용한 VC merge capable ATM switch의 경우가 보다 작은 셀 손실의 결과를 알 수 있다.

### 1.2 셀 지연

이용률의 변화에 따른 셀의 지연 시간을 비교한 결과,  $\rho=0.7$ 정도의 부하가 걸렸을 경우, VC merging capable ATM switch와 EPD 알고리즘을 이용하는 ATM switch를 비교하게 되면 많은 부하가 걸릴수록 셀의 지연시간이 비슷해진다. 그러나 RED 알고리즘을 적용한 VC merging capable ATM switch의 경우,  $\rho=0.7$ 정도에서 지연시간이 EPD 알고리즘에 비해 18% 감소함을 알 수 있다.

따라서 RED 알고리즘을 적용 할 경우 많은 부하가 걸릴수록 적은 버퍼 량이 요구되고 지연시간 또한 작다. 하지만 부하가 적을 경우 지연시간이 길어지고 그에 따른 약간의 버퍼 량이 요구된다.

## VI. 결론

본 논문에서는 망이 크게 확장 될 경우, 요구되는 Label의 수를 해결하기 위한 Merging기술 적용 시, 늘어나는 Queue length와 패킷 지연을 줄이기 위한 방법으로 RED 알고리즘을 변형하여 적용하였다. Control unit에서 폐기되어지는 셀들을 제어하고 해당 셀들을 폐기 하게된다. RED 알고리즘의 적용 시 Average queue length와 지연시간이 줄어든다는 것을 알 수 있었으나 Average queue length의 경우 링크에 Bottleneck현상이 발생하는 경우에서 시뮬레이션을 실행하였으며 지연 시간 또한 링크의 이용률이  $\rho=0.7$  이상일 경우에 성능이 좋다는 것을 알 수 있다.

따라서 링크의 효율이 낮을 경우 보다 좋은 성능을 얻기 위해 트래픽의 양이나 버스트 정도에 따라 파라미터의 값을 변형시킬 수 있는 Dynamic한

Active queue management의 적용이 요구된다.

또한 실험한 환경에서의 트래픽은 TCP만을 고려했으며 Best effort service를 가정하였으므로 Qos를 보장하기 위해 트래픽의 종류에 따라 달리 처리하는 알고리즘이 요구된다.

## 참고 문헌

- [1] Indra Widjaja and Anwar I. Elwalid, "Performance Issues in VC-Merge Capable switches for Multi-protocol Label Switching", *IEEE Communication* Vol. 17, NO. 6, pp 1178-1189 JUNE 1999
- [2] HungKei Keith Chow, A. Leon-Garcia, "VC-Merge Capable Scheduler Design" *IEEE ATM'99*, Japan, pp153-160, May 1999.
- [3] Mario Baldi, Davide Bergamasco, Silvano Gai, Dante Malagrini, "CLIMAX:Cell-Interleaved Merged ATM connection" *Telecommunication Systems* 12, 1, pp.39-49, 1999.
- [4] M. Baldi, D. Bergamasco, S. Gai, D. Malagrini, "A Comparison of ATM Stream Merging Techniques," 8th *IFIP Conference on High Performance Networking (HPN'98)*, Vienna, Austria, Sep. 1998.
- [5] I. Iliadis and P. Droz, "Impact of VC Merging on Buffer Requirements in ATM Networks", *IFIP*, 1996
- [6] Sally Floyd and Van Jacobson, "Random Early Detection Gateways for Congestion Avoidance", *IEEE/ACM Trans. on Networking*, vol.1, pp.397-413 August, 1993
- [7] B. Braden et al., "Recommendations on Queue Management and Congestion Avoidance in the Internet", *RFC 2309*, April, 1998.
- [8] S. Floyd, "Notes on Implementing RED with ATM", available from [ftp://ftp.ee.lbl.gov/email/red\\_atm.notes](ftp://ftp.ee.lbl.gov/email/red_atm.notes).

