

# Labeled OBS 망에서의 버스트 손실 감소를 위한 혼잡 제어

준회원 박 중 훈\*, 정회원 유 명 식\*

## Congestion Control for Burst Loss Reduction in Labeled OBS Network

Jonghun Park\* *Associate Member*, Myungsik Yoo\* *Regular Members*

### 요 약

오늘날 사용자들의 네트워크 사용량은 인터넷 망의 공급을 앞질러 가고 있다 따라서 단일 광섬유 하나의 잠재적인 대역폭이 큰 광 인터넷 기술은 꼭 필요한 기술로서 인식되고 그 연구가 활발히 진행되고 있다 그 중 OBS 망은 기존에 제안되었던 기술보다 상용화 가능성이 매우 크다 또한 최근에는 MPLS의 레이블 스위칭 개념을 도입한 Labeled OBS에 대한 연구가 진행되고 있다

하지만 이러한 OBS 망은 데이터를 모아서 버스트 단위로 전송하므로, 하나의 버스트 손실이라도 망 전체의 성능에 큰 영향을 미치고, 어느 한 지점으로 트래픽이 집중하게 되면 채널을 예약하지 못한 버스트들은 모두 손실로 처리되어 엄청난 양의 데이터 손실을 야기한다 따라서 OBS 망에서 혼잡 제어를 통하여 혼잡 상황이 좋아질 때까지 트래픽의 유입을 제어할 필요성이 있다

본 논문에서는 OBS 망의 혼잡 상황에 대하여 버스트 전송을 제어하는 방법을 이용하여 혼잡 상황에 효과적으로 대처하는 기법을 제안한다 제안된 혼잡 제어 기법을 통해 OBS 망의 수율 저하를 최소로 하고, 버스트의 손실은 크게 줄였다

**Key Words** : Labeled OBS, OBS, Congestion Control, Adaptive Assemble Time, Burst Assembly

### ABSTRACT

The optical Internet is considered as a feasible solution for transporting huge amount of traffic volume in the future Internet. Among optical switching technology for the optical Internet, OBS becomes one of the most promoting solution. Recently, a lebeled OBS(LOBS) architecture is considered for an efficient control on OBS network. Given that a data burst may contain few thousands of IP packets, a single loss of data burst results in a serious throughput degradation in LOBS network. In this paper, we improve the performance of LOBS network by introducing the burst congestion control mechanism. More specifically, the OBS router at the network core detects the network congestion by measuring the loss probability of burst control packet. The OBS router at the network edge reduces the burst generation according to the network condition repored by the OBS router at the network core. Through the simulations, it is shown that the proposed congestion control mechanism can reduce the burst loss probability and improve the LOBS network throughput.

\* 숭실대학교 정보통신전자공학부 (bluearts@hanmail.net, myoo@ssu.ac.kr)

논문번호 : KICS2005-06-227, 접수일자 : 2005년 06월 01일

※ 본 연구는 한국과학재단 목적기초연구R01-2002-000-00072-0(2002) 지원으로 수행되었습니다.

## I. 서론

오늘날 사용자들의 네트워크 사용량은 인터넷망의 공급을 앞질러가고 있다. 현재 전화망 트래픽은 매년 10%의 증가를 보이는 반면 인터넷 트래픽은 매년 1000%씩 증가하고 있다. 이러한 현실에서 단일 광섬유(Fiber) 하나의 잠재적인 대역폭이 50Tbps라는 장점을 이용한 광 인터넷 기술은 꼭 필요한 기술로서 인식되어 그 연구가 활발히 진행되고 있고, 기간망을 비롯하여 가입자망까지도 광을 이용한 네트워크로 진화를 시도하고 있다. 현재 광 인터넷을 실현하기 위한 기술로서 OCS(Optical Circuit Switching), OPS(Optical Packet Switching), OBS(Optical Burst Switching) 등의 3가지 방식이 논의되고 연구되고 있다[1-4].

OCS 방식은 기존의 회선교환 망처럼 파장 및 라우팅 경로를 설정한 후 데이터를 전송하는 방식이다. 하지만 OCS는 파장 변환기의 가격이 매우 비싸고 느리다는 단점이 있다. 그리고 라우팅 및 파장 할당 문제에 대한 해결책이 완전히 제시되지 못한 단점이 있다. OPS 방식은 기존 인터넷의 패킷망과 같은 개념에서 출발한다. 하지만 패킷 스위칭의 핵심은 버퍼링에 있으나 전기적인 데이터와는 달리 광 신호를 버퍼링할 수 있는 기술이 FDL(Fiber Delay Line) 외에는 특별한 대안이 없고, 광 버퍼 개발에 어려움을 겪고 있다. 또한 헤더 처리 시간 동안 데이터의 전달이 지연되는 등 데이터 패킷 스위칭에서의 문제를 그대로 갖고 있기 때문에 빠른 스위칭 시간이 요구되지만 아직 기술력에서 뒷받침되지 못하고 있다.

OBS 방식은 광 서킷 스위칭과 광 패킷 스위칭의 중간 단계로서 두 스위칭 기술의 장점을 취하고 단점을 보완하여 균형을 이루었다 할 수 있다. OBS에서는 Offset Time과 Delayed Reservation을 이용하여 광 계층에서 버퍼의 사용 없이 빠른 전송 경로의 설정이 가능하고, 네트워크 자원을 효율적으로 공유할 수 있으며, 또한 차별화된 서비스를 용이하게 제공할 수 있다. 다른 하나의 장점으로는 Offset Time을 이용함으로써 비교적 느린 스위칭 시간을 가지는 광소자를 이용하여 위에서 OPS 방식의 특징을 지니는 광 라우터를 구축할 수 있다는 점이다. 이러한 장점으로 OBS 방식은 현재 활발한 연구 개발 단계에 있다.

최근 MPLS(Multi-protocol Label Switching)의 레이블 스위칭 개념을 OBS 망에 도입한 Labeled

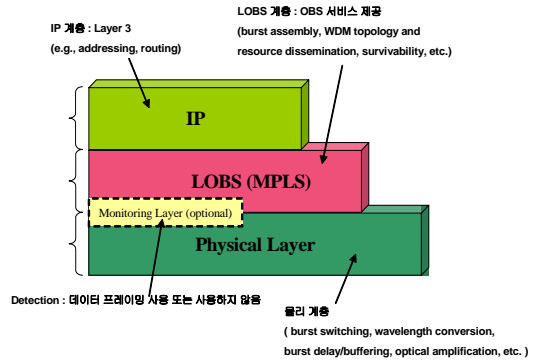


그림 1. Labeled OBS 기반의 통합된 IP-over-WDM 구조

OBS에 대한 연구가 진행되고 있다. Labeled OBS는 버스트 교환의 제어, 대역폭 예약, 그리고 가능한 FDL 버퍼의 이용과 파장 변환 등을 위하여 MPLS의 제어 메시지를 이용하는 개념이다[5-7]. 그림 1과 같은 통합된 개념의 IP-over-WDM 구조의 장점은 MPLS의 제어 방식을 이용함으로써 독립된 광 계층(OBS 계층)을 정의하고 관리하는데 복잡한 협의 과정이 필요하지 않다는 것이다. 또한 미래의 OPS 방식으로의 진화를 생각할 때, 자연스러운 과정으로 볼 수 있다.

하지만 이러한 OBS 망은 데이터를 모아서 버스트 단위로 전송하므로, 하나의 버스트 손실이라도 망 전체의 성능에 큰 영향을 미치고 어느 한 지점으로 트래픽이 집중하게 되면 채널을 예약하지 못한 버스트들은 모두 손실로 처리되어 엄청난 양의 데이터 손실을 야기한다. 따라서 OBS 망 자체도 혼잡 제어를 통하여 망의 상황이 좋아질 때까지 트래픽의 유입을 제어할 필요성이 있다.

본 논문에서는 Labeled OBS 망의 특성을 이용하여 망의 혼잡이 발생하였을 때 효율적으로 트래픽 엔지니어링을 할 수 있는 알고리즘을 제안한다. 알고리즘의 성능은 혼잡 상황이 발생하였을 경우 Ingress Edge 라우터가 버스트 전송을 얼마나 효율적으로 조절하는가와 혼잡 노드에서의 버스트 손실 비율과 손실 크기를 비교 분석하였다. 혼잡 제어 알고리즘은 AAT(Adaptive Assemble Time) 방식[8]에 기초를 둔 어셈블 방식을 사용하며, 네트워크의 혼잡 상황에 대하여 제한된 어셈블 시간과 제한된 어셈블 크기를 조절하는 것을 주요 알고리즘으로 한다.

본 논문의 구성은 다음과 같다. II에서는 본 논문에서 사용하는 AAT 버스트 어셈블 방식에 대하여

서술하고, III에서는 제안하는 Labeled OBS 망의 혼잡 제어 알고리즘에 대하여 성능을 분석한다 IV에서 시뮬레이션 환경 및 알고리즘에 대한 실험 결과에 대한 비교와 분석을 하고, 마지막으로 V에서 결론을 맺고자 한다.

## II. AAT Burst Assembly

광 버스트 스위칭에 쓰이는 버스트라는 개념은 일반적으로 같은 QoS를 지원하는 패킷들의 집합으로 정의된다. 이들 버스트를 구현하는 방법으로는 일반적으로 일정한 시간동안 같은 목적지별로 버퍼에 쌓인 데이터를 버스트로 형성하는 방법인 CAT (Constant Assemble Time)와 버스트 크기에 제한을 두어 설정된 임계값을 넘어섰을 때마다 버스트를 구현하는 CBS(Constant Burst Size)가 있다.

CAT 방식은 일정한 시간동안 같은 목적지별로 버퍼에 쌓인 데이터를 버스트로 형성하는 방법이다 따라서 CAT 방식으로 데이터를 어셈블할 경우 각각의 버스트의 크기는 서로 다르지만 버스트가 생성되어 나오는 시기는 일정하다

CBS 방식은 버스트 크기를 기준으로 버스트를 생성하는 어셈블 방법이다 이 방법은 버스트 크기에 제한을 두기 때문에 크기 설정에 따라 네트워크의 성능에 영향을 미친다 따라서 평균 생성되는 패킷 크기를 감안하여 어셈블의 기준이 되는 버스트 크기를 정하는 것이 중요하다

본 논문에서 사용하는 AAT 방식은 CAT 방식과 CBS 방식의 두 기준을 모두 포함하고 있는 방식이다[8]. AAT 방식은 일정한 시간 후에 패킷을 어셈블하여 버스트로 전송하는 방법은 CAT와 동일하지

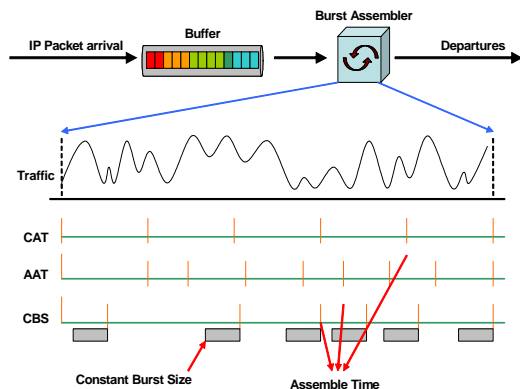


그림 2. 세 가지 유형의 버스트 형성 방법

만, 다음 어셈블 시간이 되기 전에 제한된 버스트 크기를 초과하는 경우 바로 버스트가 전송되는 점에서 CBS와 동일하다 즉, CAT와 CBS의 어셈블 방식을 모두 포함한 방법으로 적응성이 뛰어난 방식이라고 할 수 있다. 그림 2는 CAT, CBS, AAT의 세 가지 버스트 어셈블 방식을 설명하고 있다

## III. Congestion Control based on AAT (CC-AAT)

제안된 CC-AAT 알고리즘은 망의 혼잡 상황에 대해 AAT 방식의 두 가지 기준 중에서 기본적으로 설정되어 있는 버스트 크기를 적당량 감소하고, 감소된 양만큼 해당되는 버스트 전송 시간동안 버스트 생성을 정지 시키는 방법이다 CC-AAT 알고리즘은 Labeled OBS 망의 Core 부분의 혼잡 파악 부분과 Edge 부분의 혼잡 제어 부분으로 나뉜다. Core 부분은 BCP(Burst Control Packet)의 채널 예약 실패율을 통해서 발생한 혼잡 상황이 어느 정도 심각한지 판단한다. 이 판단 근거를 통해서 CC-AAT 알고리즘은 채널 예약 상황을 통하여 혼잡 상황을 인지하고 혼잡 상황의 심각성을 결정한다 결정된 혼잡 상황에 대한 정보 전달은 그림 3처럼 LABEL의 Exp 필드의 3 비트를 알맞게 설정하여 전송하며, 이를 CCI(Congestion Control Information)라고 부른다. CC-AAT는 000의 Exp 필드 값은 혼잡 제어가 이루어지지 않는 망에서의 일반적인 상태를 나타내도록 하고, 혼잡 제어 알고리즘이 구동되는 망에서는 혼잡 상태를 혼잡 정도가 상대적으로 낮은 Low 상태와 혼잡 정도가 상대적으로 큰 High의 상태로 구분하여 표시한다 혼잡 정도 Low에 대한 Exp 필드 값은 010을 설정하고, 혼잡 정도 High에 대한 Exp 필드 값은 111을 설정하였다. 이 LABEL 정보(CCI)는 다른 데이터 정보에 포함되지 않고 상태 정보값만 독립적으로 전송되거나 역방향 데이터와 함께 Piggybacking될 수 있다.

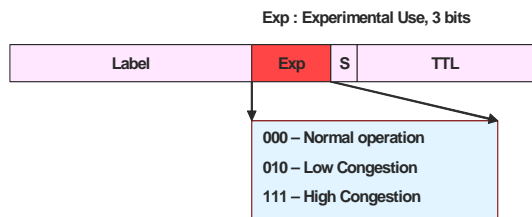


그림 3. Congestion Control Information (CCI) 설정값

### 3.1 혼잡 상황 인지

네트워크의 혼잡 상황은 혼잡 파악 기간동안 도착한 총 BCP의 수와 채널 예약에 실패한 총 BCP의 수의 비율( $r_{fail}$ )을 이용한다. 채널 예약을  $r_{fail}$ 은 다음과 같이 계산한다.

$$r_{fail} = \frac{\text{number of total BCPs that fails the channel reservation}}{\text{number of total arrivals BCP}}$$

CC-AAT는 혼잡 상황 파악 기간( $t_{CP}$ ) 동안 채널 예약 실패율  $r_{fail}$ 이 임계값의 어느 범위에 있는지를 점검하여 혼잡 정도를 파악한다. 혼잡 정도 파악은 BCP의 채널 예약이 실패하는 순간 시작되며,  $t_{CP}$  기간 동안 채널 예약 실패율  $r_{fail}$ 을 수집한다.  $t_{CP}$ 가 종료되면 혼잡 정도를 파악하여 혼잡 상황을 네트워크 Edge에 전달한다. 또한 종료된  $t_{CP}$ 는 혼잡 상황이 해소된 경우에는 완전 종료하지만, 채널 예약 실패율이 0보다 여전히 크면 그 즉시  $t_{CP}$ 를 다시 시작하여 채널 예약 실패율  $r_{fail}$ 을 수집하는 동작을 반복한다. 이 혼잡 파악 기간 동안 수집된  $r_{fail}$  정보를 이용하여  $r_{fail}$ 이 임계값의 값보다 작고 0보다 큰 경우에 혼잡 상황 Low로 판단하고,  $r_{fail}$ 이 임계값의 값보다 큰 경우 혼잡 상황 High로 판단하게 된다. 네트워크 core 부분의 혼잡 상황 파악 알고리즘을 그림 4에 요약하였다.

### 3.2 혼잡 상황 제어

CC-AAT에서 버스트를 생성하여 전송하는 Edge 부분은 CCI 정보로부터 혼잡 상황을 감지하여 그림 5에서와 같이 버스트 생성 속도를 줄이거나 증가시킨다. 버스트 생성 속도의 최대값은 미리 정의된 버스트 생성 크기 임계값과 버스트 생성 시간 임계값에 따르게 되며, 줄어든 속도는 혼잡 상황 파악 기간과는 달리 버스트 전송 속도 증가 타이머  $t_{IP}$ 를 통하여 주기적으로 증가된다. 일반적으로  $t_{IP}$ 의 기간

```

( Detect Congestion - Core part )
 $R_{fail}$  : channel reservation failure rate
 $th$  : threshold for detection of congestion state

REPEAT  $t_{CP}$  expired
  if(  $R_{fail} = 0$  )
    // Normal operation and expire  $t_{CP}$  completely
  else
    // start another  $t_{CP}$ 
    if(  $0 < R_{fail} \leq th$  )
      then Low congestion state
    else
      then High congestion state
REPEAT END  $t_{CP}$  expired
    
```

그림 4. 네트워크 코어부의 혼잡 상황 파악 알고리즘

```

( Control Congestion - Edge part )
REPEAT  $t_{IP}$  expired
  // Increase burst generation speed
REPEAT END  $t_{IP}$  expired

when CCI arrived,
  if( Exp == 010 )
    // Decrease burst generation speed Low
  else if( Exp == 111 )
    // Decrease burst generation speed High
    
```

그림 5. 네트워크 에지부의 혼잡 상황 제어 알고리즘

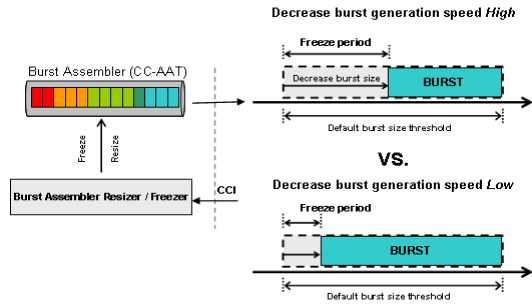


그림 6. 버스트 생성 제어 알고리즘

은  $t_{CP}$ 의 기간보다 작게 설정되며 두 기간을 초과 미터 값으로 두어 상황에 맞게 변화할 수 있다. 버스트 생성 속도 조절은 그림 6과 같이 버스트 생성 크기 임계값을 조절하는 Resizer와 동시에 줄어든 양만큼 버스트 생성 동작을 멈추도록 하는 Freezer의 두 부분을 통하여 이루어진다. Resizer는 혼잡 상태에 따라서 High의 경우에 Low의 경우보다 버스트 생성 임계값을 많이 줄여서 생성 크기를 줄인다. Freezer는 기본 임계값과 줄어든 임계값의 차이만큼의 시간동안 버스트 생성을 중지시켜서 결과적으로 줄어든 양만큼의 전송 속도 감소를 이루어낸다. 전송 속도 감소는 CCI 정보 수신에 따라서 항상 기본 임계값에 비례하여 감소하도록 하는 Non-state 방법과 이미 감소된 상태 정보를 유지하여 감소된 상태에서 비례하여 감소하도록 하는 State 방법 두 가지를 적용할 수 있다.

## IV. 성능평가 및 비교분석

### 4.1 실험 토폴로지

혼잡 제어 알고리즘에 대한 성능 평가는 그림 7과 같은 시뮬레이션 환경을 C++ 언어로 구현하여 분석하였다. 총 6개의 노드로 구성된 토폴로지는 미리 정의된 LER A → LSR E → LSR F → LER

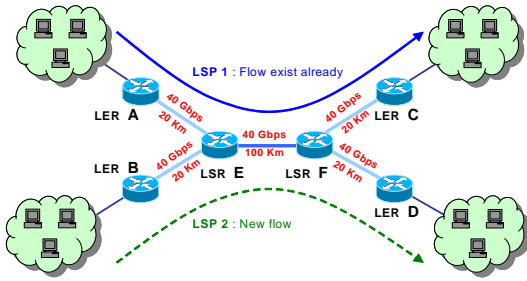


그림 7. 실험 토폴로지

C의 LSP 1과 LER B → LSR E → LSR F → LER D의 LSP 2를 통하여 버스트와 BCP가 전송된다. 각 링크는 10 Gbps의 대역폭을 갖는 4개의 채널로 구성되어있고, 생성된 데이터는 LER A와 LER B에서 버스트로 생성된다. 혼잡 제어 알고리즘의 비교는 혼잡 제어를 하지 않는 경우, 혼잡 제어는 하지 않지만 FDL을 사용하여 어느 정도 혼잡을 보상해주는 경우 그리고 FDL을 사용하여 어느 정도 혼잡을 보상해주는 혼잡 제어 알고리즘의 실험 결과를 토대로 하였다. 혼잡 상황은 이미 40 Gbps의 채널 대역폭을 거의 다 사용하고 있는 LSP 1의 플로우에 LSP 2의 플로우가 새로이 들어오면서 발생 되고, 이에 따른 CC-AAT 알고리즘의 대응 방법에 대하여 비교 및 분석하였다. 시뮬레이션 환경에서 사용한 채널 예약 알고리즘은 혼잡 상황에 대한 알고리즘의 대처 능력 파악을 위하여 단순한 First Fit 방법을 사용하였다.

시뮬레이션에서 사용된 파라미터를 표 1에 요약하였다. 평균 패킷 크기는 약 1,000 바이트로 최소 10 바이트에서 최대 1500 바이트의 패킷을 생성하여 버스트 생성을 위해 모아진다. 본 시뮬레이션 환

표 1. 실험 파라미터

파라미터	설정값
<i>avg_pck_size</i> (평균 생성 패킷 크기)	1000B
1 cycle	<i>avg_pck_size</i> 생성 시간
<i>thresh_size</i> (버스트 생성 크기 임계값)	<i>avg_pck_size</i> * 200 (약 200Kb)
<i>thresh_time</i> (버스트 생성 시간 임계값)	1 cycle * 200 * 10
<i>tIP</i> (전송 속도 증가 타이머)	1 msec
<i>tCP</i> (혼잡 상황 파악 타이머)	1 msec

경에서 평균 10000 비트의 데이터를 10 Gbps의 속도로 전송하는 경우 10 μsec의 시간이 소요된다. 이 단위를 1 사이클(cycle)로 하고 시뮬레이션의 기본 단위로 한다. 평균 패킷 부하는 시뮬레이션 시작 후에 0.85이며, 혼잡 상황이 발생하는 5 msec에는 1.7이 된다. 시뮬레이션 종료 시간은 100 msec이고, 패킷 생성은 지수 분포 방식을 따른다.

#### 4.2 시뮬레이션 결과

그림 8은 새로 유입된 플로우로 인하여 혼잡이 야기되었을 경우 기존 플로우와 새로 유입된 플로우의 버스트 생성 패턴을 혼잡 제어 알고리즘이 구동되지 않는 경우(No Congestion Control), 혼잡 상태 정보를 유지하지 않는 경우(Non-state Congestion Control) 그리고 혼잡 상태 정보를 유지하는 경우(State Congestion Control)에 대해서 나타낸 것이다. 약 5 msec부터 새로운 플로우의 전송이 시작되어 혼잡이 발생하면 각 플로우의 버스트 크기와 시간 간격은 혼잡 구간에서의 손실을 줄이도록 조절되어 전송되는 것을 볼 수 있다. LER B에서의 버스트 생성 패턴도 LER A의 5 msec 이후 패턴과 동일하게 나타난다.

그림 9는 혼잡 상황 발생에 따른 혼잡 지점에서의 버스트 손실 패턴을 나타낸 혼잡 상황에 대한 버스트 손실 비율과 손실량에 대한 변화를 100 μsec 간격으로 표시한 것이다. 혼잡 제어 알고리즘이 동작하는 경우 동작하지 않는 경우에 대비하여 버스트 손실량이 줄어들어 안정적으로 동작하는 것을 볼 수 있다. 또한 Non-state Congestion Control 방식이 State Congestion Control 방식에 비해 버스트 손실이 적은 것을 알 수 있다. 그러나 이후 그림 11

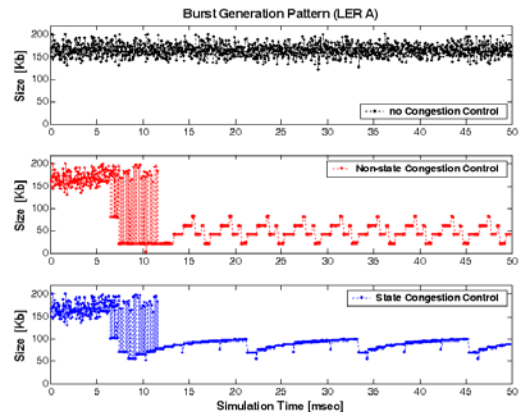


그림 8. 혼잡 상황에 대한 CC-AAT 알고리즘의 버스트 생성 패턴

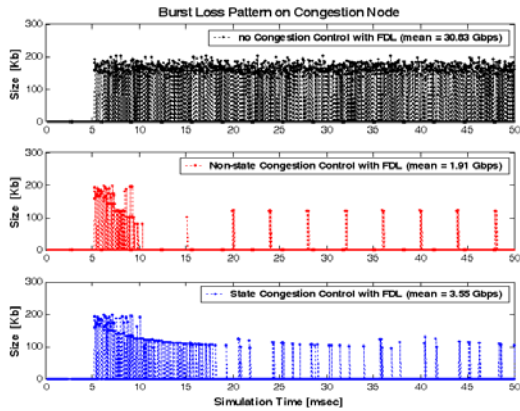


그림 9. 혼잡 지점에서의 버스트 손실 패턴

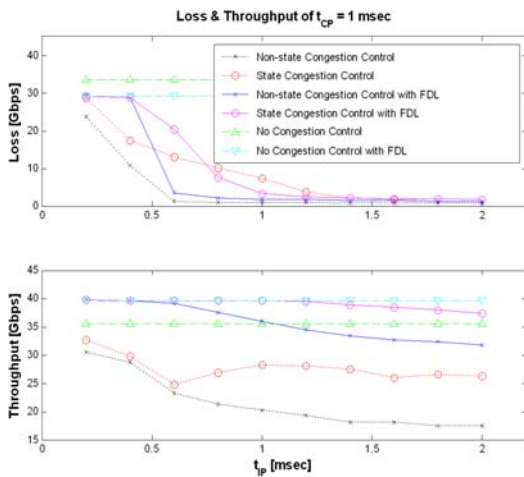


그림 10. 전송 속도 증가 타이머와 혼잡 상황 파악 타이머에 따른 알고리즘 성능 비교

에서와 같이 State Congestion Control 방식이 Non-state Congestion Control 방식 보다 높은 버스트 손실률을 갖지만 수율 측면에서는 우수한 성능을 보임을 알 수 있다.

그림 10은 혼잡 상황 파악 타이머가 1 msec인 경우에 버스트 전송 속도 증가 타이머의 값을 0.2 msec부터 2.0 msec까지 변화시켜 알고리즘의 성능 차이를 확인한 것이다. 혼잡 제어 알고리즘은 그림의 왼쪽처럼 버스트의 전송 속도 증가 시기가 빨라질수록 수율이 높은 반면 버스트의 손실 비율도 상당히 높은 것을 알 수 있다. 이는 버스트의 전송 속도 증가 시기가 점차 느려짐에 따라 손실량이 크게 줄어들게 함으로 해결 할 수 있다. 반면 너무 느린 증가 시기는 수율을 낮추는 결과를 초래하여 적당한 시기를 선택하는 것이 중요하다. 그림 10에서는

약 1~1.4 msec가 적당한 것으로 나타났다.

그림 11은 혼잡 상황 파악 타이머와 버스트 전송 속도 증가 타이머를 각각 1 msec씩 설정한 후 각 알고리즘별로 버스트의 손실 비율과 수율을 확인한 결과이다. 혼잡 제어를 하지 않는 경우에는 버스트의 손실 비율이 매우 높은 것을 알 수 있다. Non-state 혼잡 제어 알고리즘을 구동한 경우에는 버스트의 손실 비율이 상당히 감소하지만 수율의 변동 폭이 큰 것을 알 수 있다. State 혼잡 제어 알고리즘은 버스트의 손실 비율이 점차로 줄어들어 Non-state 혼잡 제어 알고리즘보다는 빈번하지만 혼잡 제어를 하지 않는 경우보다는 급격히 줄어든 것을 확인할 수 있고, 수율 또한 안정적으로 나타난다.

그림 12는 CC-AAT의 혼잡 상황 파악 임계치를 10%에서 90%까지 변경하였을 때 각 알고리즘의 버스트 손실 비율과 수율을 비교한 것이다. 임계치

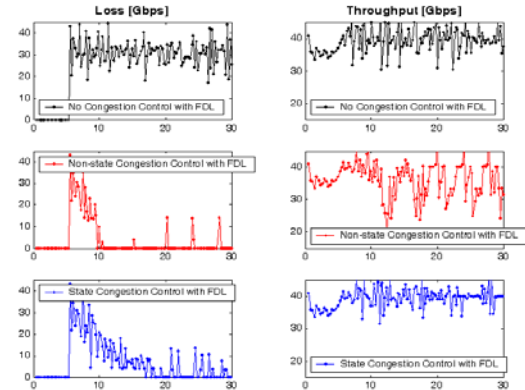


그림 11. 알고리즘별 버스트 손실 비율과 수율 비교

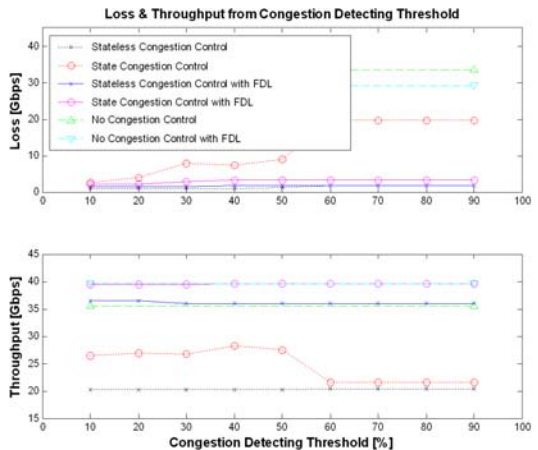


그림 12. 혼잡 상황 파악 임계치에 따른 알고리즘 성능 비교

의 변화에 따라 FDL을 사용하지 않는 혼잡 제어 알고리즘만 수율이 낮아지고 손실 비율이 높아지는 반면, 다른 알고리즘의 변화는 그리 크지 않음을 알 수 있다. 따라서 CC-AAT 알고리즘의 성능은 혼잡 상황 파악 타이머와 전송 속도 증가 타이머의 값 설정에 따라 그 성능이 좌우된다고 볼 수 있다

### V. 결론

광 스위칭 기술에 기반을 둔 광 인터넷에 대한 연구가 본격화되면서 광 인터넷을 실현하기 위한 기술로서 등장한 OBS 망은 전송한 버스트의 손실로 인한 망의 높은 비효율을 예상할 수 있다

본 논문에서는 폭증하는 인터넷의 트래픽을 수용할 수 있는 방안으로서 Labeled OBS 망의 Congestion Control based on AAT(CC-AAT)혼잡 제어 알고리즘을 제안하였다 제안한 혼잡 제어 알고리즘을 시뮬레이션을 통해 성능 분석을 하였고 이를 통하여 혼잡 제어를 하지 않는 경우에 비하여 혼잡 제어를 할 경우 버스트 손실은 감소하고, 효율은 거의 같음을 볼 수 있었다. 따라서 폭증하는 인터넷 트래픽을 OBS 기반의 광 인터넷에 수용할 때 혼잡 제어 알고리즘을 통하여 망 혼잡상황에 대한 탄력적인 대응이 필요함을 확인하였다

본 논문에서 제안한 알고리즘은 UDP 기반의 트래픽을 통하여 성능 평가 및 분석을 하였지만 TCP 등의 재전송 알고리즘을 갖고 있는 플로우 등의 경우에 더 높은 효율을 보일 것으로 예측된다

### 참 고 문 헌

[1] G. Hudek and D. Muder, "Signaling analysis for a multi-switch all-optical network," in Proceedings of Int'l Conf. on Communication (ICC), pp. 1206-1210, June 1995.

[2] D. J. Blumenthal, J. E. Bowers, L. Rau, L. Hsu-Feng Chou, S. Rangarajan, Wei Wang, and K. N. Poulsen, "Optical signal processing for optical packet switching networks," IEEE Commun. Mag. 41(2), S23-S29, 2003.

[3] C. Qiao and M. Yoo, "Optical Burst Switching (OBS) - A New Paradigm for an Optical Internet," Journal of High Speed Networks, vol. 8, no. 1, pp. 69-84, Jan. 1999.

[4] L. Xu, H. G. Perros, and G. Rouskas,

"Techniques for Optical Packet Switching and Optical Burst Switching", IEEE Communications Magazine, Vol. 39, pp. 136-142, Jan., 2001.

[5] C. Qiao and M. Yoo "Choices, Features and Issues in Optical Burst Switching (OBS)," Optical Net. Mag., vol. 1, no. 2, Apr. 2000, pp. 36-44.

[6] C. Qiao and M. Yoo, "Optical Burst Switching(OBS)-A New Paradigm for an Optical Internet," Journal of High Speed Networks, vol. 8, no. 1, pp. 6-84, Jan. 1999.

[7] Chunming Qiao, "Labeled Optical Burst Switching for IP-Over-WDM Integration," IEEE Comm. Mag., vol. 38, Sept. 2000. pp. 104-114.

[8] Dong-Hee Park, M. Yoo, "Comparative study on burst assembly schemes for optical burst-switched networks," Proc. SPIE, vol. 4910, pp. 266-274, Sept, 2002.

박 종 훈 (Jonghun Park)

준회원



2003년 2월 숭실대학교 전자공학과 졸업  
2005년 2월 숭실대학교 정보통신공학과(석사)  
<관심분야> 인터넷 혼잡 제어, 인터넷 QoS 보장, 광네트워킹, OBS

유 명 식 (Myungsik Yoo)

정회원



1989년 2월 고려대학교 전자전산공학과 졸업  
1991년 2월 고려대학교 전자공학과(석사)  
2000년 8월 Dept. of Electrical Engineering, SUNY at Buffalo(박사)  
2000년 9월~현재 숭실대학교 정보통신전자공학부 조교수  
<관심분야> 광네트워킹, OBS, 인터넷 QoS, Wireless TCP, Wireless Link Protocol