

# MPLS 망에서 확장성을 갖는 RSVP-TE 프로토콜 구현을 위한 메시지 부하의 영향 분석

이 영 우\* · 박 재 형\*\* · 김 상 하\*\*\*

## 요 약

인터넷의 고품질 서비스를 제공하기 위한 기술로 MPLS 트래픽 엔지니어링 기술이 대두되고 있다. MPLS 망에서 트래픽 엔지니어링은 서비스 품질을 보장할 수 있는 경로를 계산하고 MPLS 신호 프로토콜을 이용하여 망 자원을 예약하는 기능을 수행한다. MPLS 신호 프로토콜의 하나인 RSVP-TE 프로토콜은 폴로우 기반으로 설정된 경로의 연결 상태 관리를 위해 주기적으로 Refresh 메시지를 송수신한다. 이러한 주기적인 메시지의 송수신은 프로토콜 메시지 처리 부하를 증가시키기 때문에 많은 수의 경로 설정 능력을 제한하는 문제점을 유발한다. 본 논문은 이러한 RSVP-TE 프로토콜의 Refresh 메시지의 부하를 줄이기 위하여 제시된 Refresh Reduction 기법의 영향에 대하여 분석한다. 또한, 확장성을 갖는 RSVP-TE 프로토콜을 구현하기 위해서 메시지 부하가 적은 기법을 제안한다.

## Evaluation on Effect of Message Overhead for Implementing a Scalable RSVP-TE Protocol in MPLS Networks

Young-Woo Lee\* · Jaehyung Park\*\* · Sang-Ha Kim\*\*\*

## ABSTRACT

For providing high quality-guaranteed services over Internet, traffic engineering based on an MPLS technology is being introduced. MPLS traffic engineering performs the computation on the path guaranteeing service's quality and the reservation on network resources by an MPLS signaling protocol. As one of MPLS signaling protocols, RSVP-TE protocol transmits and receives periodic refresh messages for maintaining the path of a traffic flow. Such characteristic gives a heavy processing overhead to routers for maintaining states of large number of paths. In this paper, we propose a scalable implementation approach for RSVP-TE without dramatically increasing processing overhead. And we evaluate the processing overhead on periodic messages by implementing the RSVP-TE protocol and the reduction mechanism of periodic messages.

**키워드 :** MPLS 망(MPLS Network), RSVP-TE, 확장성(Scalability), Refresh 감소(Refresh Reduction), Hello 프로토콜(Hello Protocol)

## 1. 서 론

최근 인터넷의 응용 서비스로 VoIP와 비디오 스트리밍과 같은 실시간 응용 서비스가 등장함에 따라서 인터넷에서 고품질을 보장할 수 있는 기능이 요구되고 있다. 이러한 요구는 네트워크에서 트래픽을 분류 처리하고 자원을 효율적으로 관리하기 위한 서비스의 품질(Quality of Service) 보장 기술과 트래픽 엔지니어링 기술을 필요로 한다[4, 5, 8]. 한편, 인터넷의 트래픽 처리 능력의 고속화와 새로운 서비스의 수용을 위해서 제안된 MPLS(MultiProtocol Label Switching) [10] 기술은 인터넷의 고품질 서비스를 제공하기 위한 기술로 대두되고 있다. MPLS는 패킷 헤더에 부착된 레이블을 이용한 스위칭에 의해 패킷을 전달하여 고속의 패킷 전달을

지원한다. 또한, 비연결형 네트워크인 IP망에 연결형의 경로인 LSP(Label Switched Path)를 설정함으로써 명시적인 경로 설정과 트래픽 엔지니어링 기능을 지원할 수 있는 특성을 가지고 있다.

인터넷에서 고품질 서비스를 제공하기 위한 트래픽 엔지니어링 기법은 제어 측면과 데이터 전달 측면으로 구분된다. 제어 측면에서는 대역폭, 지연시간, 손실 등의 QoS 정보를 근거로 서비스 품질을 만족시키는 경로를 계산하는 QoS 라우팅과 QoS 정보를 경로 상에 있는 모든 라우터에게 전달하여 자원을 예약하는 신호 프로토콜이 존재한다. 한편, 데이터 전달 측면에서는 해당 서비스에서 생성된 트래픽을 구분하여 예약된 자원 내에서 전달될 수 있도록 하는 기능이 존재한다. 특히, MPLS 기술에 기반한 트래픽 엔지니어링 기법은 신호 프로토콜로서 CR-LDP(Constraint-based Route Label Distribution Protocol)나 RSVP-TE(Resource Reservation Protocol for Traffic Engineering)를 이용하여 품질

\* 정 회 원 : KT 운용시스템연구소 사업기획부장

\*\* 중 심 회 원 : 전남대학교 전자컴퓨터정보통신공학부 교수

\*\*\* 중 심 회 원 : 충남대학교 전기정보통신공학부 교수

논문접수 : 2004년 6월 19일, 심사완료 : 2004년 9월 2일

보장형 LSP에 대해 설정/해제/관리를 수행한다[1, 6].

MPLS 신호 프로토콜 중의 하나인 RSVP-TE는 특정 서비스나 가입자의 품질 요구사항이나 운용자의 QoS 정책을 고려한 조건 기반의 LSP를 지원할 수 있도록 RSVP를 확장한 프로토콜이다. RSVP-TE 프로토콜은 명시적인 경로 정보와 레이블 정보 및 트래픽 플로우 정보 전달을 통하여 품질 보장형 LSP에 대한 설정/해제/관리를 수행한다. RSVP-TE는 IP 기반 메시지 송수신 방법과 Soft-State 방식의 LSP 관리 방법을 사용하므로 LSP 별로 연결 상태를 관리하기 위해 주기적인 Refresh 메시지를 송수신한다. 이와 같이 주기적으로 전달되는 프로토콜 메시지는 라우터의 메시지 처리 부하를 증가시키기 때문에 많은 수의 LSP를 지원하는 대규모 MPLS 망에 적용하기 어려운 단점이 있다[3].

RSVP-TE의 확장성의 문제를 해결하기 위하여 주기적으로 전달되는 메시지의 수를 줄이는 Refresh 메시지 감소 기법이 제안되었다. 본 논문에서는 Refresh 메시지 감소 기법이 RSVP-TE 프로토콜의 확장성에 미치는 영향을 분석한다. 또한, 프로토콜의 Hello 기법을 확장하여 많은 수의 LSP를 관리하는데 메시지 처리 부하가 적은 RSVP-TE의 구현 방법을 제시한다. 그리고, 주기적인 메시지 처리 부하 측면에서 RSVP-TE 프로토콜의 성능을 평가한다.

본 논문의 구성은 다음과 같다. 2장에서는 RSVP-TE 프로토콜과 Refresh 메시지 감소 기법에 대해서 기술하고, 3장에서 Refresh 메시지 감소 기법의 구현 구조를 설명하고 확장성을 갖는 RSVP-TE 구현을 위한 구조를 제시한다. 4장에서는 기존 Refresh 기법과 Refresh 메시지 감소 기법에 따른 프로토콜 메시지의 처리 부하를 분석하고 확장성을 갖는 RSVP-TE 구조에 대해서 평가하고, 5장에서 결론을 맺는다.

## 2. RSVP-TE

본 장에서는 MPLS 신호 프로토콜의 하나인 RSVP-TE에 대해서 기술하고, 프로토콜의 확장성을 높이기 위해서 제시된 Refresh 메시지 감소 기법에 대해서 기술한다.

### 2.1 RSVP-TE 프로토콜

인터넷에서 실시간 응용 서비스를 제공하기 위하여 네트워크의 자원을 예약할 수 있는 프로토콜로서 RSVP가 설계되었다[2, 7, 11]. 서비스에서 요구하는 자원을 예약하기 위해서 RSVP 프로토콜은 Path와 Resv 메시지를 송수신하여 서비스에서 생성된 트래픽을 전송할 경로(Tunnel)를 설정한다. 또한, 설정된 경로의 상태를 관리하기 위해서 동일한 Path/Resv 메시지를 주기적으로 송수신하는 Soft State 관리 기법을 수행한다. 이와 같은 주기적인 메시지를 Refresh 메시지라 칭한다.

현재 네트워크를 구성하는 대부분의 라우터는 UNIX 기반의 운영체제로 구현된다. UNIX 기반 시스템에서 메시지의 크기보다는 메시지의 개수가 UNIX 소켓 입출력에 큰 영향을 미치는 것으로 확인되었다[3]. 즉, UNIX 소켓 입출력에

소요되는 처리 부하는 크기가 큰 메시지와 적은 메시지에 차이가 없으나, 많은 수의 메시지가 소켓에 입출력될 경우 혼잡이 발생하여 처리 부하가 높아진다. 특히, 설정된 경로에 대한 Soft State 관리 기법이 적용되는 RSVP 프로토콜은 플로우가 많을수록 많은 수의 Refresh 메시지를 생성하므로 경로상의 라우터에 메시지 처리 부하를 증가시킨다. 이러한 이유로 RSVP 프로토콜은 소규모의 네트워크에 적용되었다.

MPLS 기능은 인터넷 상에서 서비스 품질을 보장할 수 있는 방법으로, 서비스 품질을 보장하기 위해서 QoS 관련 정보를 전달하는 신호 프로토콜로서 RSVP를 확장한 RSVP-TE 프로토콜이 설계되었다. RSVP에 명시적인 경로 설정과 MPLS 레이블을 전달할 수 있는 개체를 추가하여 LSP를 설정 및 해제하는 기능을 확장하였다. 그러나, LSP에 대한 연결 상태 관리 기법은 기존 RSVP와 같은 Soft State 관리 기법을 이용하기 때문에 많은 수의 LSP에 대해서는 메시지 처리 부하가 큰 단점을 지닌다.

한편, MPLS 망에서 RSVP-TE 프로토콜에 의해서 설정된 LSP 경로의 상태 관리는 Refresh 메시지를 송수신함으로써 이루어진다. 하지만, Refresh 메시지는 설정된 LSP를 통해서 전달되는 것이 아니고, 경로상의 모든 라우터를 지나서 IP 경로를 통해서 전달된다. 즉, Refresh 기간 내의 설정된 LSP의 상태를 관리하는 기능 보다는 오히려 경로상의 IP 경로에 상태를 관리하는 것이다. MPLS 망에 적용된 RSVP-TE 프로토콜은 LSP에 대한 관리가 별도로 요구된다.

### 2.2 Refresh 감소 기법

RSVP-TE를 이용하여 QoS 품질 보장형 서비스를 실현하는데 있어서 MPLS 망에서 대부분의 메시지는 Refresh 메시지이다. 왜냐하면, QoS 조건을 만족하는 LSP를 설정하는 메시지는 초기 연결을 위해서 보내지고, LSP가 설정되고 난 후에는 해당 경로에 대한 변경이 거의 없으므로 LSP의 연결 상태를 관리하는 Refresh 메시지가 주기적으로 전달된다. 그러므로, 네트워크에서 많은 수의 LSP를 지원하기 위해서 주기적인 Refresh 메시지의 개수를 줄이는 방법이 제시되었다[3].

Refresh 감소 기법은 많은 수의 LSP를 지원하여 확장성을 높이고, RSVP-TE의 메시지의 신뢰적인 전송을 지원하는 방법이다. 제시된 Refresh 감소 기법은 크게 세 가지의 기능으로 구성되었다. 첫 번째는 Acknowledgement 메시지의 사용과 재전송 기능으로 메시지 전달에 있어서 신뢰적인 전송을 지원하는 기능이다[9]. 두 번째 기능은 다수 개의 RSVP 프로토콜 메시지를 하나의 메시지로 묶어서 전달하는 방법(Bundling)으로 전달되는 메시지의 수를 줄임으로써 메시지 처리 부하를 줄일 수 있다. 세 번째는 설정된 각각의 LSP에 대하여 라우터에서 유일한 ID로 표현하고, 표현된 여러 개의 ID를 메시지로 묶어서 전달하는 Summary Refresh (Srefresh) 기능이다. Srefresh 기능은 Path 또는 Resv 메시지를 전달하는 기존 Refresh 메시지와 달리 유일한 ID만을 전달하는 것으로 첫 번째 기능인 Bundling 기법보다 많은 수의 LSP에 대한 상태를 관리할 수 있는 장점이 있다.

### 3. 확장성을 갖는 RSVP-TE 구현 구조

본 장에서는 MPLS 망에서 확장성을 갖는 RSVP-TE 프로토콜을 구현하기 위해서 Hello 프로토콜을 확장하는 방법을 제시하고, 확장성을 갖는 RSVP-TE 프로토콜의 구현 구조에 대해서 기술한다.

#### 3.1 Hello 프로토콜의 확장

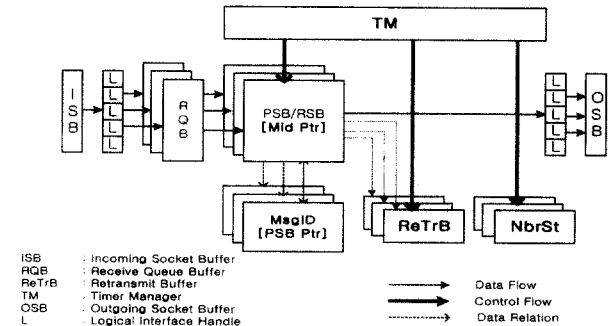
MPLS 망에서 RSVP-TE 프로토콜에 의해서 설정된 LSP의 연결 상태 관리는 Refresh 메시지를 주기적으로 전달함으로써 수행된다. 그러나, Refresh 메시지가 IP 경로상에 있는 라우터들로 전달됨으로써 LSP에 대한 연결 상태는 장애 처리 기능과 같은 별도의 기능이 요구된다. 그러므로, IP망과 달리 MPLS 망에서 Refresh 메시지는 설정된 LSP에 대한 연결 상태를 관리하는 기능으로 수행되지 못하고 경로상의 라우터들이 RSVP-TE 세션이 올바르게 동작하고 있는지를 관리하는 기능을 수행한다. 임의의 라우터와 이웃 라우터간에 RSVP-TE 세션의 동작을 관리하기 위해서 Hello 프로토콜을 확장한다.

Hello 프로토콜을 통해서 라우터간의 물리적 또는 논리적인 연결에 대한 장애와 라우터의 장애에 대한 정보를 검사할 수 있도록 한다. 이와 같은 라우터의 상태 관리 기능을 RSVP-TE가 동작하는 이웃하는 두 라우터간에 주기적으로 Hello 메시지를 주고 받음으로써 수행되며, Hello Request 객체와 Hello Acknowledgement 객체를 송수신한다. 이웃 라우터의 장애가 발생하는 경우에는 해당 인터페이스로 설정된 모든 LSP에 대해서 LSP의 시작 라우터와 LSP의 끝 라우터로 연결 오류 메시지를 전달하는 기능을 추가한다.

#### 3.2 RSVP-TE 프로토콜 구현 구조

RSVP-TE 프로토콜 엔진의 전체적인 구조는 (그림 1)에서와 같이 타이머 관리 블록, Path/Resv 상태 관리 블록, 이웃 라우터 상태 관리 블록으로 구성된다. LSP가 설정되면, 프로토콜 엔진에서는 Path/Resv 상태관리(PSB/RSB) 블록에 설정된 LSP에 대한 정보를 저장하고, 타이머 관리(TM) 블록은 Refresh 기법을 수행하는 동작으로 주기적인 프로토콜 메시지를 전송한다. 이웃 라우터 상태관리(NbrSt) 블록은 Hello 메시지를 처리하여 이웃 라우터와 연결 상태를 검사하는 것으로, 통신 장애가 발생하여 이웃 라우터와 Hello 메시지가 제대로 송수신 되지 않을 경우 해당 라우터를 지

나는 모든 LSP에 대해서 장애 메시지를 생성하여 LSP의 시작과 끝 라우터로 전송한다.



(그림 1) RSVP-TE 프로토콜 엔진의 구조

신뢰적인 프로토콜 메시지의 전송과 Refresh 메시지의 감소에 대한 기법을 적용하기 위해서 프로토콜 엔진은 메시지의 번호를 관리하는 MsgID 테이블과 재전송을 위한 메시지 버퍼인 ReTrB가 존재한다. MsgID 테이블은 RSVP-TE 프로토콜에서 새로운 LSP 설정할 때 부여하며, 해당 LSP에 대한 Path/Resv 메시지를 관리하는 PSB/RSB에 참조 포인터를 갖는다. 참조 포인터에 의해서 Summary Refresh인 경우 해당 LSP에 대한 상태를 PSB/RSB에서 관리할 수 있다. ReTrB는 전송된 프로토콜 메시지에 대한 Acknowledgement를 받지 못한 메시지를 저장하고 있는 임시의 버퍼로서 일정 시간 경과 후에도 Acknowledgement 메시지가 오지 않는 경우 같은 메시지를 전송한다.

MPLS 망에서 LSP 설정 및 관리를 지원하기 위해서 RSVP-TE 프로토콜 구현에 적용된 기본적인 자료구조는 (그림 2)에서 보여준 바와 같이 PSB/RSB 테이블과 논리적 인터페이스 처리 테이블이다. PSB/RSB 테이블은 시작 라우터 주소, 끝 라우터 주소, LSP ID와 플로우 ID로 구성된다. Refresh 감소 기법을 적용하기 위해서 MsgID 부분이 추가되었다. 논리적 인터페이스 처리 테이블은 LSP의 흐름을 구하는 것으로 바로 전 라우터와 다음 라우터의 정보를 관리하여, 그러한 라우터로 Refresh 메시지를 전송할 때 이용된다.

#### 3.3 RSVP-TE 프로토콜 메시지 처리 구현

RSVP-TE 프로토콜 기능을 수행하기 위해서는 다음과 같은 3종류의 메시지 처리를 지원하여야 하므로 본 논문에서는 각각의 기능을 구현한다.

PSB/RSB 테이블	
시작 라우터 주소	IP Address 값
끝 라우터 주소	IP Address 값
LSP ID	0 ... (2 <sup>16</sup> - 1)
Flow ID	0 ... (2 <sup>16</sup> - 1)
입출력 인터페이스	논리 인터페이스 인덱스
MsgID	메시지 ID 참조 포인터

논리적 인터페이스 처리 테이블(LIH)	
상태	0 ... (2 <sup>8</sup> - 1)
자기 주소	IP Address 값
논리 인터페이스 인덱스	0 ... (2 <sup>32</sup> - 1)

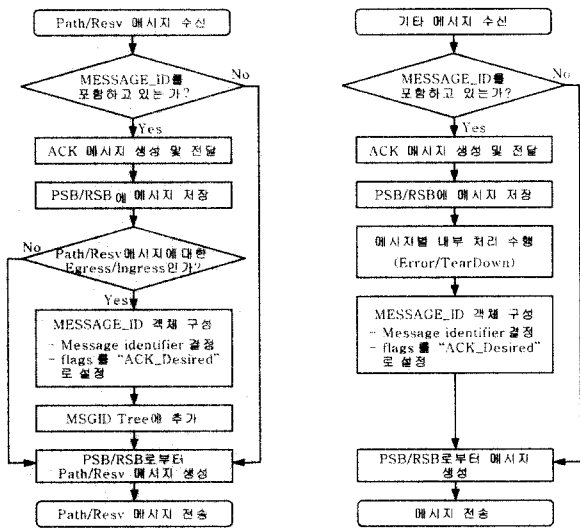
(그림 2) RSVP-TE 구현의 기본적인 자료구조

3.3.1 MESSAGE\_ID 메시지 처리 기능

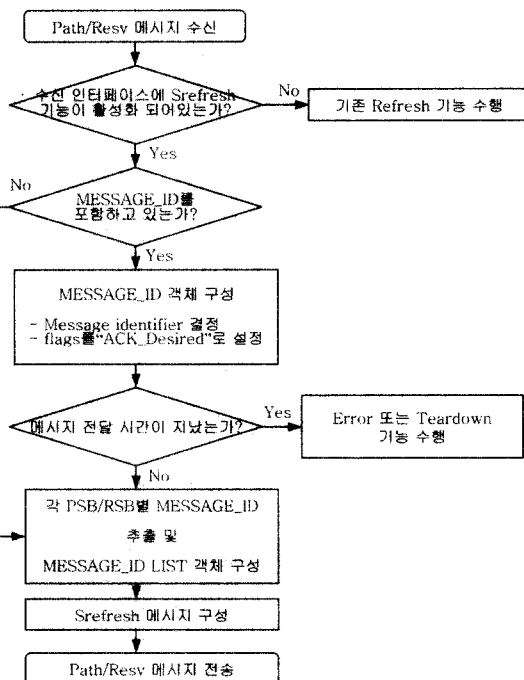
MESSAGE\_ID 객체가 포함된 RSVP 메시지 수신시 MESSAGE\_ID 객체의 flag값이 "ACK\_Desired"로 설정된 경우 MESSAGE\_ID\_ACK 객체를 포함한 ACK 메시지로 응답한다.

이 때, MESSAGE\_ID\_ACK 객체의 flag는 "ACK\_Desired"로 설정하고, msg\_identifier 값을 수신된 MESSAGE\_ID 객체의 msg\_identifier를 복사하여 사용한다.

(그림 3)은 MESSAGE\_ID 객체가 포함된 메시지를 수신하는 경우의 흐름도를 보여준다.



(그림 3) MESSAGE\_ID 객체를 포함 메시지의 수신 절차



(그림 4) Srefresh 메시지 구성 및 전송 절차

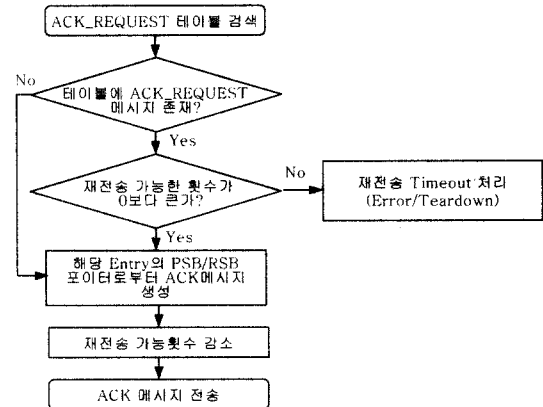
3.3.2 Summary Refresh 메시지 처리 기능

Path/Resv Refresh 메시지를 대신하는 Srefresh 메시지 적

용하면 기존의 MESSAGE\_ID객체를 포함한 Path/Resv메시지 송수신에 의하여 저장된 msg\_identifier값을 이용한다. Srefresh 메시지를 생성하는 과정은 PSB테이블을 검색하여 이미 저장된 msg\_identifier를 순차적으로 검색하여 MESSAGE\_ID 객체의 msg\_identifier 필드에 추가하여 구성한다. (그림 4)는 Srefresh 메시지를 구성하고 전송하는 과정을 보여준다. Srefresh 메시지 수신 및 처리 과정은 msg\_identifier 값을 확인하고 각 entry별로 MsgID 테이블로부터 검색하여 해당 PSB의 상태를 변경한다.

3.3.3 재전송 기능

MESSAGE\_ID 객체의 "ACK\_Desired" flag를 포함한 메시지에 대하여 ACK 메시지의 응답이 없으면 동일한 메시지를 재전송 한다. 이를 위하여 송신 노드는 MESSAGE\_ID 객체의 "ACK\_Desired" flag를 사용하는 메시지를 담을 수 있는 MSG\_ACK\_REQUEST 테이블을 구성하여 ACK 메시지의 응답 여부를 확인하고, 응답이 없으면 설정된 시간 내에 재전송 한다. (그림 5)는 재전송 기능을 위한 처리 절차를 보여준다.



(그림 5) 재전송 처리 절차

4. 성능 평가

본 장에서는 MPLS 망에서 LSP의 상태를 관리하기 위한 주기적인 Refresh 메시지가 라우터의 성능에 미치는 영향을 분석한다. 또한, Hello 프로토콜을 확장하여 구현한 RSVP-TE 프로토콜에 대한 성능을 평가한다.

4.1 실험 환경

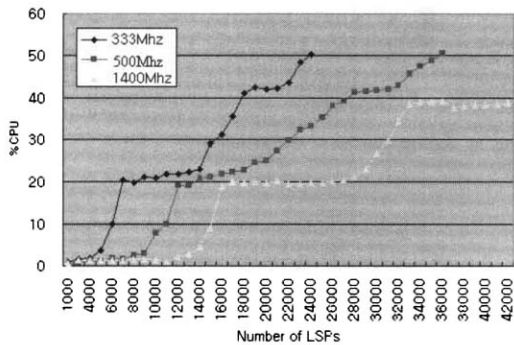
본 논문에서 RSVP-TE 프로토콜의 Refresh 메시지의 부하를 측정하기 위해서 Linux를 기반으로 구현된 라우터에서 RSVP-TE 프로토콜을 동작시키고, 설정된 LSP 개수에 따라 기존의 Refresh 기법과 Summary Refresh 기법에 대하여 CPU 부하를 각각 측정하였다. 또한, LSP 설정을 위한 초기 메시지 송수신 및 처리시에 발생하는 CPU 부하는 제외하고, 설정된 이후의 Refresh 및 Srefresh 송수신 및 처리에 대한 CPU 부하만을 측정하였다. Refresh 메시지 Interv-

al은 30초로 설정하고, Multiplier는 3회로 설정하였다. 즉, Refresh Timeout이 되는 기준은 90초 동안 Refresh 메시지가 수신되지 않는 경우 해당 LSP의 연결 상태가 끊어진 것으로 간주된다.

#### 4.2 Refresh 메시지 처리 부하

(그림 6)은 MPLS 망에서 RSVP-TE 프로토콜을 적용하였을 경우 기존의 Refresh 기법에 대한 메시지 처리 부하를 보여준다. (그림 6)에서 보는 바와 같이 333Mhz CPU에서는 24000개 이상의 LSP를 설정하는 경우 Refresh Timeout이 발생하였으며, 500Mhz CPU에서는 36000개 이상의 LSP를 설정하는 경우 Refresh Timeout이 발생하였고, 1400Mhz CPU에서는 42,000개 이상의 LSP 설정하는 경우 Refresh Timeout이 발생하는 것을 볼 수 있다.

Timeout 발생하는 것은 Refresh 메시지가 90초 내에 1개라도 수신하지 못하는 경우이므로, Refresh 메시지를 처리하는 시간이 많아져서 이웃 라우터로 메시지를 전송하지 못하는 상황이다.

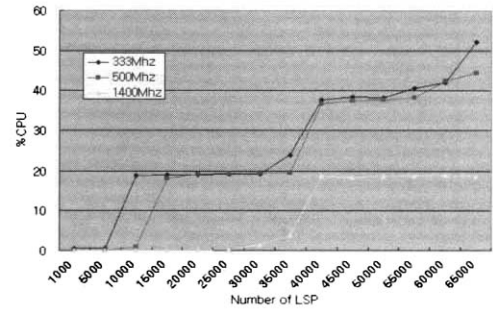


(그림 6) 기존 Refresh 기법을 적용한 RSVP-TE의 메시지 처리 부하

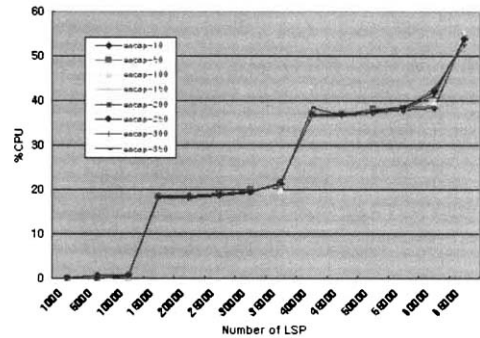
한편, Refresh 감소 기법을 적용한 RSVP-TE 프로토콜의 Srefresh 메시지를 처리 부하는 (그림 7)에서와 같이 보여진다. Timeout 되지 않으며 설정 가능한 LSP 수는 기존 Refresh 기법을 적용한 RSVP-TE 프로토콜에 비해서 증가함을 알 수 있다. 그러나, 어느 정도 이상의 LSP가 설정된 이후에는 CPU 부하가 급격히 증가하고 있음을 나타내고 있다. Srefresh 메시지를 사용한다는 것은 송수신 되는 메시지 수를 줄임으로서 프로토콜의 메시지 처리 부하를 줄이는데 효과가 있으나 CPU 부하가 급격히 줄지 않음을 알 수 있다. 그러한 이유는, ACK 응답 메시지가 새롭게 추가되었으며, 많은 수의 MsgID 값을 테이블에서 찾는 과정을 수행하여야 하기 때문이다.

Refresh 감소 기법은 여러 개의 MsgID를 하나의 메시지에 묶어서 보낼 수 있는 기능을 제공한다. (그림 8)은 하나의 메시지에 묶는 MsgID의 개수에 따른 메시지 처리 부하를 보여준다. (그림 8)에서와 같이 설정된 LSP 수에 따라 하나의 메시지에 묶어진 MsgID 수는 CPU의 메시지 처리 부하에 큰 영향을 주지 않음을 볼 수 있다. 즉, RSVP-TE

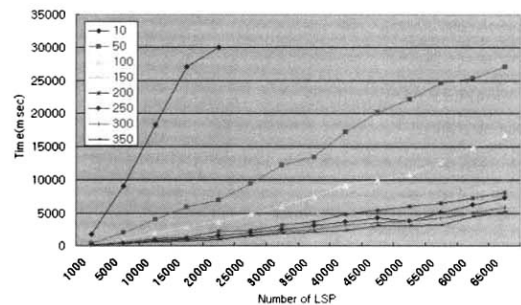
프로토콜의 메시지 처리 부하는 설정된 LSP 수에 영향을 받기 때문이다. (그림 8)은 500Mhz CPU에서의 메시지 처리 부하를 보여주는 것으로 하나의 메시지로 묶는 MsgID의 수를 10, 50, 100, 150, 200, 250, 300, 350으로 변경하면서 얻은 결과이다.



(그림 7) Refresh 감소 기법을 적용한 RSVP-TE의 메시지 처리 부하



(그림 8) MsgID의 개수에 따른 메시지 처리 부하

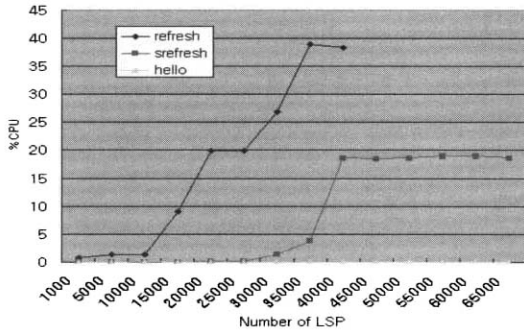


(그림 9) MsgID의 개수에 따른 메시지 전송에 소요되는 시간

#### 4.3 Hello 프로토콜을 확장한 RSVP-TE의 메시지 처리 부하

MPLS 망에서 RSVP-TE의 Refresh 기법은 설정된 LSP의 연결 상태를 관리하기 위한 목적으로 제시되었지만, 설정된 LSP 경로 상에 존재하는 라우터들의 상태만을 관리할 수 있다. Hello 프로토콜을 확장함으로써 라우터들의 상태를 관리할 수 있다. (그림 10)은 RSVP-TE 프로토콜에 대해서 Refresh 기법들과 Hello 프로토콜을 적용하였을 경우에 메시지 처리 부하를 보여준다. 이 때, 1400Mhz의 CPU를 사용하는 Linux에 기반한 라우터에서 얻은 결과이다. Hello 프로토콜을 확장

한 경우에는 LSP의 개수에 거의 영향을 받지 않는 결과를 보이므로 MPLS 망에서 확장성을 갖는 RSVP-TE 프로토콜을 구현할 수 있다.



(그림 10) Hello 프로토콜을 적용한 RSVP-TE 프로토콜의 메시지 처리 부하

### 5. 결 론

인터넷 상에서 사용자의 서비스 품질을 보장하기 위해서 트래픽 엔지니어링 기술이 개발되어왔으며, 특히 MPLS는 트래픽 엔지니어링 기술을 쉽게 접목시킬 수 있는 기술로 적용되고 있다. 트래픽 엔지니어링을 위한 MPLS 신호 프로토콜로서 RSVP-TE는 서비스 품질을 요구하는 LSP를 설정하는 프로토콜로 사용되고 있으나, Refresh 기법으로 인해서 확장성에 단점을 가지고 있다. 본 논문에서는 RSVP-TE 프로토콜의 Refresh 기법이 라우터의 CPU에 미치는 영향에 대해서 평가하였다. Refresh 기법은 주기적인 프로토콜 메시지를 생성시켜서 라우터의 메시지 처리 부하를 증가시켰으며, 또한 Refresh 감소 기법도 메시지 처리 부하를 크게 감소시키지 못함을 보였다. 본 논문에서는 메시지 처리 부하를 증가시키지 않고 라우터의 상태를 관리할 수 있는 Hello 프로토콜을 확장하는 방법을 제시하였다. 제시된 방법은 MPLS 망에서 확장성을 갖는 RSVP-TE 구현에 적용할 수 있다.

### 참 고 문 헌

[1] D. Awduche, L. Berger, D. Gan, T. Li, V. Srinivasan, and G. Swallow, "RSVP-TE: Extensions to RSVP for LSP Tunnels," IETF RFC3209, December, 2001.

[2] R. Braden, L. Zhang, S. Berson, S. Herzog and S. Jamin, "Resource ReSerVation Protocol (RSVP) - Version 1 Functional Specification," IETF RFC 2205, September, 1997.

[3] L. Berger, D. Gan, G. Swallow, P. Pan, F. Tommasi, and S. Molendini, "RSVP Refresh Overhead Reduction Extensions," IETF RFC 2961, April, 2001.

[4] D. Clark, S. Shenker, and L. Zhang, "Supporting Real-Time Applications in an Integrated Services Packet Network: Architecture and Mechanism," ACM SIGCOMM '92, 1992.

[5] T. Ishihara, T. Kusuda, G. Nakami, I. Nakagawa, Y. Kikuchi and H. Esaki, "A Consideration of IX Architecture

Using MPLS based on Router Performance and QoS Requirements," IEICE Trans. On Communications, Vol. E86-B, No.2, pp.498-505, February, 2003.

[6] B. Jamoussi, et al., "Constraint-based LSP Setup Using LDP," IETF RFC 3212, January, 2002.

[7] G.-S. Kuo and P.-C. Ko, "Dynamic RSVP," IEEE Communications Magazine, pp.130-135, May, 2003.

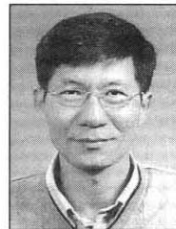
[8] E. Osborne and A. Simha, "Traffic Engineering with MPLS," Cisco Press.

[9] P. Pan and H. Schulzrinne, "An Evaluation on RSVP Transport Mechanism," draft-pan-nsis-rsvp-transport-01.txt, Internet Draft, February, 2003.

[10] E. Rosen, A. Viswanathan and R. Callon, "Multiprotocol Label Switching Architecture," IETF RFC 3031, January, 2001.

[11] L. Zhang, S. Deering, D. Estrin, S. Shenker, and D. Zappala, "RSVP: A New Resource ReSerVation Protocol," IEEE Communications Magazine, pp.116-127, May, 2002.

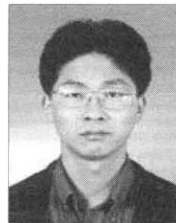
### 이 영 우



e-mail : ywlee@kt.co.kr  
 1984년 숭실대학교 전자공학과(학사)  
 1990년 숭실대학원 전자공학과(석사)  
 1990년~2003년 KT 연구개발본부  
 선임연구원  
 2004년~현재 KT 운용시스템연구소  
 사업기획부장

관심분야 : MPLS TE, IP QoS, NMS, Convergence Network

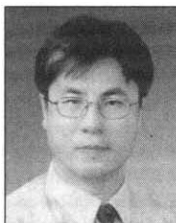
### 박 재 형



e-mail : hyeoung@chonnam.ac.kr  
 1987년~1991년 연세대학교 전산학과  
 (학사)  
 1991년~1993년 KAIST 전산학과(석사)  
 1993년~1997년 KAIST 전산학과(박사)  
 1997년~1998년 KAIST 인공지능연구소  
 Post-Doc 연수연구원

1998년~2002년 ETRI 네트워크연구소 선임연구원  
 2002년~현재 전남대학교 전자컴퓨터정보통신공학부 조교수  
 관심분야 : MPLS, 인터넷 프로토콜, 인터넷 시스템, 라우터 구조,  
 인터넷 네트워크 연결망, 병렬처리, 멀티캐스트

### 김 상 하



e-mail : shkim@cnu.ac.kr  
 1980년 서울대학교 화학과(학사)  
 1984년 U. Of Houston computer science  
 (석사)  
 1989년 U. Of Houston computer science  
 (박사)

1990년~1991년 한국과학기술원 선임연구원  
 1992년~현재 충남대학교 전기정보통신공학부 교수  
 관심분야 : IP QoS, Mobile Network, Multicast, Optical Internet