

論文2003-40TC-10-15

ns-2 네트워크 시뮬레이터 기반의 MPλS 시뮬레이터의 설계 및 구현

(Design and Implementation of MPλS Simulator based on ns-2 Network Simulator)

徐善英*, 李奉煥*, 黃善台*, 尹贊鉉**

(Seon-Young Seo, Bong-Hwan Lee, Suntae Hwang, and Chan-Hyun Youn)

요약

인터넷 사용자의 급증, 서비스의 다양화, 전송속도의 고속화, 이용 분야의 확대 등으로 인해 인터넷 트래픽이 매년 급속하게 증가하고 있다. 이러한 수요를 충족시키기 위하여 대용량의 대역폭을 제공하는 WDM 기술과 IP 망을 효율적으로 통합한 MPLS를 기반으로 하는 MPλS 프로토콜이 제안되었다. 본 논문에서는 ns-2 네트워크 시뮬레이터의 기능을 확장하여 MPλS 네트워크 시뮬레이터를 설계하고 구현하였다. 구현한 시뮬레이터는 OXC(Optical Crossconnect) 모듈, 효율적인 파장 할당을 위한 RWA(Routing and Wavelength Assignment) 모듈, 다중 파장 링크 모듈, MPλS 시그널링 및 제어 모듈 등으로 구성된다. 구현한 시뮬레이터 상에서 처리율 및 블로킹 확률 등의 파라미터를 이용하여 MPλS의 성능을 검증하였으며, 구현한 시뮬레이터는 시스템 개발 이전 단계에서 제안된 프로토콜의 기능을 검증하는데 유용하게 사용될 수 있을 것으로 사료된다.

Abstract

The rapid increase of Internet users, diversity of application services, high speed data transmission, and extension of application areas have resulted in exponential growth of Internet traffic. In order to satisfy the increasing demand of bandwidth, the MPλS protocol, which is based on MPLS for efficient integration of WDM and IP protocols, has been suggested by IETF. In this paper, we present an MPλS simulator which enables to simulate various MPλS function such as optical crossconnect (OXC), multi-wavelength links, routing and wavelength assignment(RWA), and MPλS signaling and control. The simulator is developed based on the ns-2, an widely used multi-protocol network simulator. The function of the simulator is validated by running many simulation based on various scenarios and performance measures such as throughput and blocking probability. The simulator could be widely utilized for validation of proposed protocols before developing real optical network systems.

Keywords : Simulator, MPλS, WDM, OXC, IP

* 正會員, 大田大學校 情報通信工學科

(Department of Information and Communications Engineering, Daejeon University)

** 正會員, 韓國情報通信大學院大學校 工學部

(School of Engineering, Information and Communi-

cations University)
※ 본 연구는 한국과학재단 광인터넷 연구사업(ERC) 지원으로 수행되었음.

接受日字:2003年9月7日, 수정완료일:2003年10月10日

I. 서론

매년 급속도로 증가하는 인터넷 트래픽으로 인하여 인터넷과 광 전송망에 대한 관심이 두드러지고 있으며, 증가하는 트래픽의 수요를 충족시키기 위하여 대용량의 대역폭을 제공하는 광인터넷 기술이 개발되고 있다. 네트워크의 병목현상은 주로 전송링크와 스위칭 노드의 용량 부족, 그리고 서버의 처리용량 부족에 기인한다. 네트워크 전송링크의 용량 부족에 의한 병목현상은 최근의 기술발전으로 수십 테라비트에 이르는 전송용량을 제공할 수 있는 WDM(Wavelength Division Multiplexing) 전송기술로 해결할 수 있다. 현재 가능한 스위칭 노드의 용량 또한 1Tbps 정도까지 확장이 가능한 것으로 전망되고 있으나 이것이 증가하는 망 트래픽의 증가율을 따라갈 수 있을지의 여부에 대해서는 불투명한 상태이다^[1]. 그러므로 IP망과 WDM 전송기술 사이의 효율적인 통합을 위한 네트워크 기술이 요구된다. 이를 위해 WDM 기술에 MPLS(Multi-Protocol Label Switching)를 기반으로 한 MPλS 접근 방식을 결합한 광 전달망 구조가 제안되었다.

MPλS(Multi-Protocol Lambda Switching) 네트워크는 채널 단위의 파장(lambda)을 MPLS의 레이블(label) 개념으로 처리함으로써 새로운 제어평면을 설계할 필요 없이 기존의 MPLS를 수정하여 사용하는 개념이다^[2]. MPLS가 IP와 WDM 사이의 통합 구조로 사용되는 이유는 MPLS가 트래픽 엔지니어링을 제공하고, 레이블을 파장으로 바꾸어 효과적으로 광 네트워크로 확장할 수 있다는 장점 때문이다^[2,3]. 즉, 시그널링에 사용되는 신호 표준 중의 하나인 MPLS의 LDP(Label Distribution Protocol)를 MPλS 네트워크의 시그널링을 위해 사용할 수 있다. 그러나 현재 많은 분야에서 광인터넷을 위한 MPλS 기술의 연구가 진행되고 있지만, 이러한 연구를 뒷받침할 시뮬레이터가 개발되지 않아 MPλS 네트워크의 성능과 기능을 평가하는데 어려움이 있다.

본 논문의 목적은 MPλS 네트워크 환경에서 광인터넷 구성요소 및 MPλS의 제어기능을 시험할 수 있는 시뮬레이터를 개발하여 MPλS와 관련된 연구의 타당성을 검증할 수 있도록 하는 것이다. 본 연구에서 구현한 MPλS 시뮬레이터는 현재 네트워크 시뮬레이터로 널리 사용되고 있는 ns-2(Network Simulator v2)^[4]와 MPLS 시뮬레이터^[7]의 기능을 확장하여 구현하였다. ns-2에는

노드, 링크뿐만 아니라 트래픽 모델과 여러 인터넷 프로토콜들이 구현되어 있고, NAM(Network Animator)이라는 애니메이션 도구를 사용하여 시뮬레이션의 결과 및 패킷의 추적을 GUI로 볼 수 있도록 해준다. 구현한 MPλS 시뮬레이터는 ns-2의 기능을 기반으로 파장 스위칭, 광전송 링크, 파장할당 모듈, 라우팅 및 파장 제어를 위한 시그널링 기능을 포함한다. 구현한 시뮬레이터를 이용하여 다양한 네트워크 토폴로지에 대한 네트워크 파라미터들을 설정한 다음 처리율 및 블로킹 확률 등을 이용하여 MPλS의 기능을 시험하고, 그 결과를 NAM과 XGraph로 표현하여, 성능을 평가하고 분석하였다.

본 논문의 구성은 I 장의 서론에 이어서, II 장에서는 MPλS 기술이 등장하기까지의 광 네트워크 발전과정 및 본 논문의 기반 기술인 MPLS와 MPλS 네트워크의 기능 및 특징에 대하여 자세히 기술하고, ns-2의 네트워크 환경을 소개한다. III 장에서는 구현한 MPλS 시뮬레이터의 구조와 설계 및 구현에 대하여 설명하고, IV 장에서는 구현한 MPλS 시뮬레이터를 사용하여 다양한 시뮬레이션을 통하여 나온 결과에 대해 분석하였다. 마지막으로 V 장에서 본 논문의 결론을 맺는다.

II. 관련 연구

1. 광인터넷 기술

인터넷 사용자의 급증으로 인해 음성 트래픽 보다 데이터 트래픽의 비중이 훨씬 커지면서 기존의 정보통신망 구조가 급격히 재편되고 있다. 지금까지 네트워크의 주력장비로 사용된 ATM과 SDH 장치 기능을 간략화시켜, WDM 장치 위에 바로 인터넷 트래픽을 싣도록 하는 IP over WDM 기술이 본격적인 개발 단계에 접어들었다. IP over WDM 구조하에서 가장 중요한 이슈는 WDM으로 IP 데이터그램을 가장 경제적이면서 효율적으로 전송함과 동시에 확장이 매우 용이한 네트워크 구조에 대한 것이다.

현재 전달망으로 가장 유력한 형태는 광회선 스위칭 네트워크이다. 대역폭 활용에 있어서 비효율적인 광회선 스위칭 네트워크의 성능을 향상시키기 위해 필요한 기술은 복수개의 파장 자원을 어떻게 동적으로 연결하여 트래픽을 전달하는 광경로(lightpath)를 구성할 것인가이다. 이를 위해 등장한 기술이 MPLS의 제어평면을 새롭게 변화시킨 MPλS 네트워크이다^[1].

2. MPLS 프로토콜

IETF는 일본 도시바의 레이블 스위칭, Ipsilon사에서 제안한 IP 스위칭 기술, Cisco사에 의해 제안된 태그 스위칭의 개념들을 정리하고 표준화하여 1977년 3월 MPLS^[3,6] 워킹 그룹을 구성하였다. MPLS는 입력되는 패킷에 있는 레이블 필드가 다음 노드를 결정하는데 사용되도록 하는 교환 방법이다. 각 노드에서 입력 레이블은 다음 노드에서 사용되는 다른 레이블로 변환된다. 이렇게 설정되어지는 경로를 LSP(Label Switching Path)라 하고, 입력 레이블을 이용하여 다음 노드로의 포워딩을 결정하는 라우터를 LSR(Label Switched Router)이라 한다.

LSR 노드는 패킷의 레이블을 사용하여 LSP 경로를 설정하며 포워딩과 라우팅 절차를 단순화 및 고속화한다. LSR 노드는 패킷의 입력 레이블을 보고 적절한 처리 절차를 결정한다. MPLS LSP 경로는 두 가지 방법으로 지정된다. 하나는 운용자가 다양한 정책과 트래픽 관리를 실현하기 위해 사전에 경로를 지정하는 명시적 라우팅이고, 다른 하나는 정규적인 IP 라우팅 경로를 MPLS 서브넷 상에 오버레이하는 홉 라우팅이다. IP 패킷이 MPLS 도메인을 통과하기 위해서는 MPLS Shim 헤더의 레이블을 인덱스로 사용하여 다음 노드를 결정하고, FEC(Forwarding Equivalence Class)를 정의하여 패킷들을 특정 LSR에서 목적지가 같은 IP 패킷들 또는 같은 서비스로 처리되는 패킷들은 하나의 FEC로 분류하여 전달하며, 레이블 된 패킷을 수신하면 더 이상 3계층의 정보를 분석할 필요 없이 레이블 된 패킷의 레이블 값만을 검사하여 패킷 전송 처리를 수행하므로 네트워크의 처리속도를 빠르게 할 수 있다. 현재의 IP 네트워크는 best effort 서비스만 가능하기 때문에 서비스 등급에 따라 차등화 된 실시간 서비스를 위해서는 flow 단위로 트래픽을 구분하여 전달해야 한다. 또한 평균 1000bit 정도로 매우 가변적인 인터넷 패킷을 10Gbps 이상의 광대역의 WDM 채널을 사용하여 전송 할 경우, 동일한 목적지로 향하는 같은 등급의 메시지 흐름을 하나의 파장으로 통합하여 태그(tag)를 붙여 보내면 채널 효율이 높아지고, 서비스 등급별로 차등적인 전달이 가능하다. 이러한 방법 중의 하나가 MPLS이다^[6].

1999년 11월에 IETF에서는 IP 라우터와 OXC를 함께 사용함으로써 완벽하게 통합된 네트워크를 구축하기 위해 MPLS 프로토콜을 확장하는 MPLS가 제안되었고^[2],

이것은 라우팅과 시그널링을 위해 기존의 프로토콜을 사용할 수 있다는 장점을 가지고 있다. MPLS 네트워크는 MPLS의 트래픽 엔지니어링과 WDM의 전송 능력을 결합시킨 방식으로 MPLS의 레이블 대신에 파장(λ)을 할당하여 사용하는 개념이다. 즉, MPLS 네트워크의 OXC의 데이터 평면은 입력포트에서 출력포트로 광경로를 제공하기 위하여 각 OXC는 토폴로지 정보를 기반으로 생성된 라우팅 테이블을 토대로 스위치 매트릭스와 파장 테이블을 구축하고 이것을 참조하여 입력 포트를 출력 포트에 스위칭하고, 입력 파장을 출력 파장으로 변환하여 다음 노드로 패킷을 전송한다.

3. ns-2 네트워크 시뮬레이터

MPLS 시뮬레이터는 ns(Network Simulator)^[9,10] 시뮬레이터를 기반으로 구현되었다. ns-2는 TCP, 라우팅 프로토콜, 멀티캐스트 프로토콜, RTP(Real Time Protocol), SRM(Scalable Reliable Multicast) 등 다양한 인터넷 프로토콜에 대한 시뮬레이션을 수행하기에 적절한 여러 환경을 제공하는 네트워크 시뮬레이션 도구이다.

<그림 1>은 ns-2를 이루는 네트워크 구성요소들이다. OTcl은 Tcl 스크립트 언어에 객체 지향 개념을 추가한 언어이고, Tccl은 OTcl과 C++언어를 서로 링크시키기 위해 제공되는 언어이다.

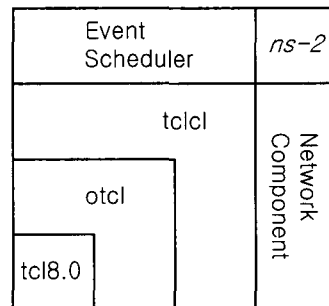


그림 1. ns-2 시뮬레이터의 구조^[11]
 Fig. 1. Architecture of ns-2 simulator^[11].

시뮬레이션 처리 시간을 줄이기 위하여 이벤트 스케줄러와 데이터 경로에 있는 기본적인 네트워크 구성요소 객체는 C++를 사용하여 작성하고 컴파일 되어 패킷의 움직임 및 상태정보의 관리를 위해 사용되고, OTcl 언어는 시뮬레이션을 위한 이벤트를 설정할 때 사용한다. ns-2를 이용하여 시뮬레이션 결과를 표현하고 분석

하기 위하여 <그림 2>에 나타난 것과 같이 VINT 프로젝트의 일부로 개발된 NAM(Network Animator)과 XGraph가 사용된다. NAM은 VCR에서 제공하는 것과 같은 play, fast forward, rewind, pause 기능을 GUI 형태로 제공하며, 디스플레이 속도를 조절할 수 있는 기능을 가진 Tcl/tk로 작성된 도구이다. 즉, 네트워크 시뮬레이션 도중 발생하는 사건들을 시간 축에 대해 기록해 놓은 파일 정보를 분석하여 그 결과를 화면상에 애니메이션 형태로 표현하는 비주얼 시뮬레이션 도구이다. 다른 도구로는 시뮬레이션 결과를 분석하여 그 결과를 시각적으로 화면에 표현하는 XGraph가 있다. XGraph는 점이나 선, 막대그래프 등의 형태로 네트워크 시뮬레이션 결과를 표현한다. XGraph는 시뮬레이션 과정에서 만들어진 추적 파일로부터 원하는 노드 사이의 패킷 흐름을 선택하여 그래프로 표현하는 것이 가능하다.

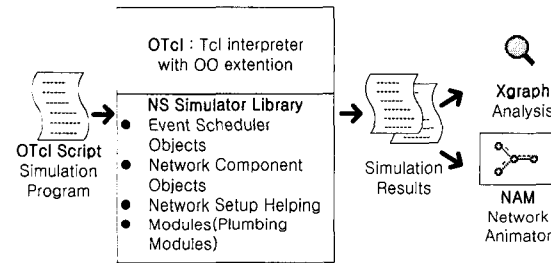


그림 2. ns-2에서의 시뮬레이션 처리 과정
Fig. 2. Simulation procedure in ns-2.

III. MPλS 시뮬레이터의 설계

본 연구에서는 ns-2에서 기본적으로 제공하는 네트워크 요소들을 사용하여 MPλS 네트워크의 구성 요소들을 구현하였다. 파장스위칭 및 파장변환 기능이 존재하는 MPλS 노드를 구현하고, 최단 경로 알고리즘에 기반한 라우팅과 효율적인 파장 할당을 위한 RWA (Routing and Wavelength Assignment) 모듈과 MPLS의 LDP 시그널링 프로토콜을 확장하여 MPλS에서 파장 분배를 담당하는 WDP 에이전트(Wavelength Distribution Protocol Agent)를 구현하였다.

설계한 MPλS 시뮬레이터에서 각 노드는 <그림 3>에서와 같이 크게 제어 평면과 데이터 평면으로 구분된다. 제어 평면(Control Plane)은 MPλS 네트워크에서 라우팅 정보에 기반하여 파장 할당을 담당하는 시그널링 부분으로 MPLS의 제어 평면을 확장하였다.

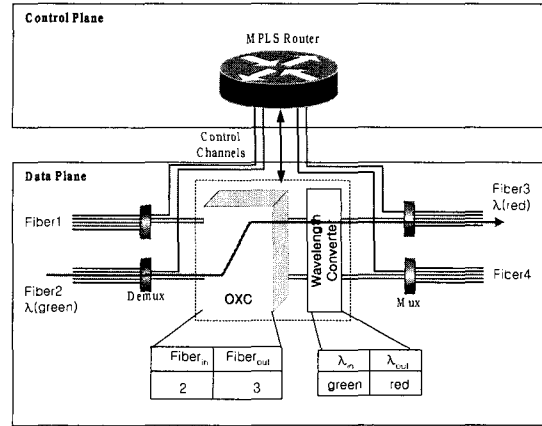


그림 3. MPλS 시뮬레이터의 구성 요소
Fig. 3. Components of MPλS Simulator.

데이터 평면(Data Plane)은 실제로 데이터의 포워딩을 담당하는 부분이며, 입력 포트를 출력 포트로 변환하는 OXC 모듈과 제어 평면의 시그널링을 통해 얻은 파장 정보를 이용하여 파장 테이블을 생성하고 입력파장을 출력파장으로 변환하는 WC(Wavelength Converter) 모듈로 구성되어 있다. 각 노드에서는 라우팅에 기반한 패킷의 포워딩과 관련하여 파장 관리 테이블을 생성하고, 현재 사용중인 파장은 busy, 사용 가능한 파장은 free로 표시한다. 구현한 MPλS 시뮬레이터에서는 네트워크를 구성하는 각 노드에 파장 변환기가 존재한다고 가정하여 모든 노드에서 파장변환을 하는 것으로 구현하였다. <그림 3>의 MPλS 노드의 구조를 ns-2 컴포넌트를 이용한 설계도를 상세히 나타내면 <그림 4>와 같다.

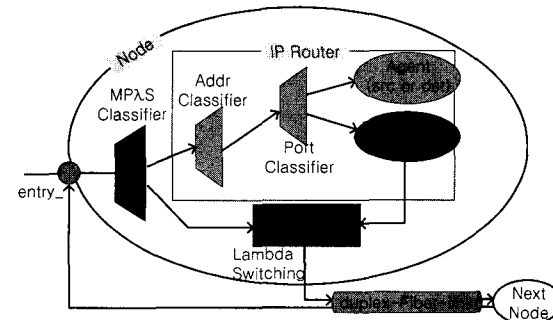


그림 4. MPλS 노드의 구조
Fig. 4. MPλS node architecture.

구현한 ns-2 시뮬레이터에서 하나의 노드는 기본적으로 에이전트(Agent)와 분류자(Classifier)로 구성되어 있

다. <그림 4>에서 에이전트의 기능은 프로토콜의 트래픽을 생성하여 전송하는 송신자 역할과 도착한 패킷을 처리하는 수신자에 관련된 역할을 하는 개체이며 분류자는 다음 노드로 전달하기 위한 패킷의 분류를 담당하는 개체이다. Addr. Classifier는 패킷의 목적지 주소에 기반한 포워딩을 담당하며, 만약 패킷의 목적지가 자기 자신이라면 Port Classifier에게 전달하고, 그렇지 않으면 패킷은 파장 변환을 거쳐 다음 노드로 전달된다. 시뮬레이터의 기본 노드에 새로 추가된 MPλS Classifier는 파장을 할당받은 패킷인지를 분류하여 파장할당을 받은 패킷이라면 다음 노드로 전달하기 위하여 노드를 빠져나가기 전에 파장 변환기에 의하여 다른 파장으로 파장 스위칭(Lambda Switching)이 이루어진다. 파장을 할당받지 못한 패킷이라면 Addr. Classifier로 전달된다. WDP Agent는 파장을 할당하기 위한 시그널링 패킷을 생성하거나 처리하는 역할을 담당한다. 또한, 현재 사용 가능한 파장과 사용되고 있는 파장의 수를 포함하는 테이블을 관리하고 있으며, 광경로 요청, 생성, 수정 및 해제와 같은 시그널링을 처리한다.

<그림 5>는 ns-2 네트워크 시뮬레이터에서 제공하는 컴포넌트들을 사용하여 MPλS 시뮬레이터를 구현하는 과정을 나타낸 것이다. 즉, ns-2에서 제공하는 node, agent, classifier 클래스를 이용하여 MPλS node, agent, classifier 등을 구현하였다. node와 agent는 OTcl로 구현하였으며 classifier와 파장 할당 관리자(WAssignManager)는 C++로 구현하였다.

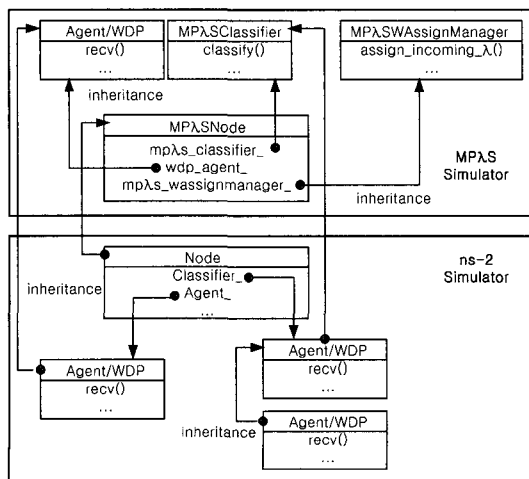


그림 5. MPλS와 ns-2 시뮬레이터 컴포넌트간의 상관관계
Fig. 5. Relationship between MPλS and ns-2 simulator components.

노드와 노드를 연결하는 duplex-fiber-link는 ns-2에서 제공되는 duplex-link를 사용한 양방향 전송 링크이며, MPλS 네트워크에서는 광 데이터를 실어 나르는 광링크(Fiber link)라 가정하였다. WDM 기술을 이용하면 하나의 광링크는 여러 개의 파장을 가질 수 있으므로 초당 기가 비트 단위의 전송속도를 가지는 여러 채널을 수용할 수 있다. 이에 따른 광링크의 최대 대역폭은 다음과 같다.

$$\text{파장당 최대대역폭} = \frac{\text{각링크의 대역폭}}{\text{링크당 파장의수}} \quad (1)$$

NAM에서는 다중 파장을 갖는 광링크를 표현할 수 없으며 실제로 광 패킷의 전송 과정을 시뮬레이션 과정에서 광링크의 대역폭의 크기에 비해 광패킷의 크기는 매우 작기 때문에 광패킷의 흐름을 디스플레이 하는데 어려움이 따른다. 따라서 구현한 시뮬레이터에서는 위의 식과 같이 광링크의 파장 당 최대 대역폭은 초당 메가 비트의 전송속도를 가진다고 가정하였다.

본 연구에서 구현한 MPλS 시뮬레이터는 위에서 언급했듯이 모든 OXC들이 파장 변환 능력을 가진 것으로 가정한다. 만약 파장 변환 능력이 없다면 근원지와 목적지 노드 사이에 같은 파장이 모든 링크에서 사용될 것이다. 이것은 링크에 사용 가능한 파장이 있는 경우에도 연결 요청이 블로킹됨을 의미하며 망의 성능을 저하시킨다. 블로킹 확률은 중간 OXC에서 파장변환(wavelength conversion) 즉, 하나의 파장을 또 다른 파장으로 교환하는 연결 능력에 따라 감소될 수 있다. 일반적으로 파장 변환 이득은 네트워크 토폴로지, 광섬유당 파장의 개수, 링크 당 광섬유의 개수, 트래픽 부하, 라우팅과 파장 선택 알고리즘과 같은 많은 요소에 의존한다. 특히 광섬유에서 지원되는 파장의 수가 증가함에 따라, 파장의 효율을 개선하기 위해 물리적 망상에서 광경로를 어떻게 라우팅하며 어떤 파장을 광경로에 할당할 것인가에 대한 문제로 인하여 RWA는 더욱 중요하게 되었다^[12].

광 네트워크를 위해 연구되어 온 라우팅과 파장 선택 알고리즘(RWA)은 휴리스틱 알고리즘과 최적 알고리즘의 관점에서 연구되어 왔으며 다음과 같이 분류할 수 있다^[13, 14].

■ 라우팅 알고리즘

(1) Fixed Shortest Path Routing algorithm: 사전에 결

정된 최단 경로를 사용하여 연결을 설정하는 라우팅 방식이다. 경로배정이 간단하며 네트워크 상에 모든 근원지-목적지 노드간에 최소 비용경로 알고리즘이 사용되어 경로가 미리 계산되고, 각 노드의 라우팅 테이블에 저장된다.

- (2) Alternate Routing algorithm: 고정된 순서에 따라 순차적으로 사용 가능한 대체 경로의 집합을 구성하고 라우팅한다.

■ 파장 선택 알고리즘

- (1) First-Fit: 파장은 오름차순으로 인덱싱(indexing)된다. 사용 가능한 파장들의 집합 중에서 가장 낮은 인덱스를 갖는 파장을 골라서 할당한다.
- (2) Random: 사용 가능한 파장 집합 중에서 임의로 파장을 할당한다. 이 알고리즘은 모든 파장에 부하(load)를 분산시키는 성질을 가지고 있어서 각각의 파장은 거의 동일한 이용률을 나타낸다.
- (3) Most-used: 전체 망에서 가장 많이 이용되는 파장을 골라서 할당한다.

본 논문에서 구현한 MPλS 시뮬레이터의 라우팅을 위해 ns-2가 제공하는 최단 경로 라우팅(Shortest Path Routing) 알고리즘을 사용하였고, 파장할당을 위해서는 사용 가능한 파장 중에서 가장 낮은 인덱스를 갖는 파장을 우선 할당하는 First-Fit의 파장 선택 알고리즘을 구현하여 시뮬레이션하였다.

IV. 시뮬레이션 시나리오 및 결과

1. 시뮬레이션 환경 및 시나리오

MPλS 시뮬레이터는 레드햇 리눅스(Redhat Linux) v6.1의 시스템에서 ns 버전 2.1인 ns-2.1b6을 기반으로 Tcl, OTcl 스크립트와 C++언어를 사용하여 개발되었다. 시뮬레이션을 위해서 실제로 사용되는 언어는 Tcl 스크립트이다. 시뮬레이션을 위한 환경을 Tcl 스크립트로 프로그래밍하고 시뮬레이션 결과를 NAM이나 XGraph로 보여준다. MPλS 시뮬레이터의 기능에 대한 성능을 평가하기 위하여 <그림 6, 7, 8>과 같은 세 종류의 네트워크 토폴로지를 사용하였다. <그림 6>은 vBNS(very high performance Backbone Networks Services) 토폴로지로서 연결 상태가 강하지 않은 대표 네트워크라 할

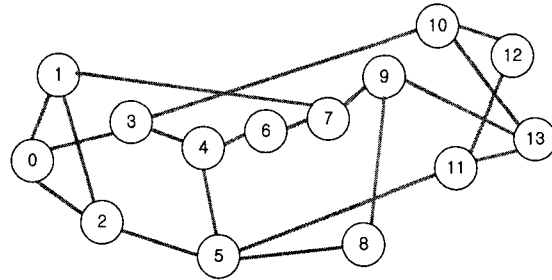


그림 6. vBNS의 NSFnet 백본 네트워크^[1]
Fig. 6. vBNS Backbone network topology^[1].

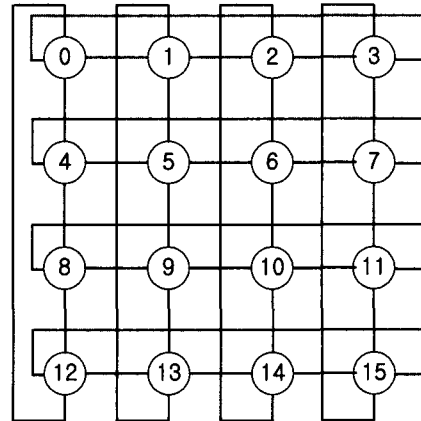


그림 7. 4x4 mesh-torus 네트워크^[13]
Fig. 7. 4x4 mesh-torus topology^[13].

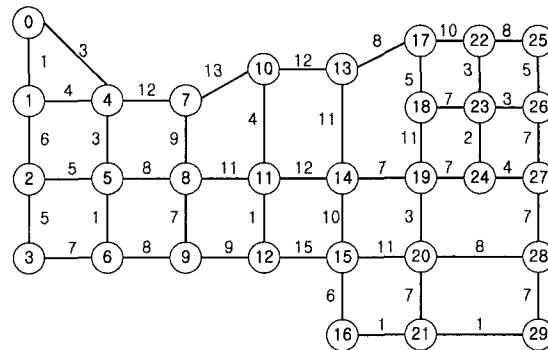


그림 8. 30 노드 메쉬 네트워크^[14]
Fig. 8. 30 node mesh network topology.

수 있으며, 네트워크 기술 개발 연구 및 시험에 주로 사용되는 네트워크이다. 한편, <그림 7>의 4x4 메쉬-토러스 망은 강한 연결 상태를 가진 망 형태를 대표하는 네트워크 구조이다. <그림 8>은 NSFNet이나 4x4 메쉬형보다 큰 네트워크 구조를 시뮬레이션 하기 위한 30노드 메쉬 네트워크이다. 특히, <그림 8>은 <그림 6> 및

<그림 7>과는 달리 노드와 노드 사이에 서로 다른 링크 비용을 가지는 경우로서 동일한 링크비용을 가지는 네트워크와의 성능을 비교 시험해보고자 한다.

시뮬레이션을 하기 전에 <그림 6>과 <그림 7>의 토폴로지에 대하여 동일한 부하를 가했을 때 파장의 수가 늘어날 때 처리율의 변화를 예측하고 부하가 점차적으로 늘어날 때 블로킹 확률의 추이를 예측하여 예측 결과와 실제 시뮬레이션 결과와의 비교를 통하여 시뮬레이터 구현의 적정성을 판단하고자 한다. 4x4 메쉬형 네트워크는 vBNS 네트워크 보다 단단히 연결되어 있기 때문에 근원지에서 목적지로의 경로가 더 많아지게 된다. 즉, vBNS 네트워크는 메쉬형 네트워크에 비하여 노드에서의 시간지연이 길어지고 경로의 부족으로 인하여 처리율이 낮아지게 될 것이다. 한편, 블로킹 확률의 측면에서 보면 메쉬형 네트워크의 블로킹 확률이 부하가 올라갈수록 상대적으로 더 커진다. 즉, vBNS 네트워크는 16개의 링크와 91개의 O-D(Origin-Destination) 쌍을 가지고 있으며 메쉬형 네트워크는 32개의 링크와 120개의 O-D 쌍을 가지고 있다. 두 개의 네트워크를 상대적으로 비교하면 링크의 수가 배로 늘어날 때 O-D 쌍의 수는 약 1.3배만 늘어나므로 메쉬형 네트워크가 상대적으로 더 높은 블로킹 확률을 보일 것으로 예측된다. 한편, 16 노드 메쉬형 네트워크와 30 노드 메쉬형 네트워크를 비교하면 30 노드 메쉬형 네트워크는 각 링크가 서로 다른 링크 비용을 가지므로 근원지에서 목적지로의 경로가 상대적으로 적어지게 된다. 따라서 16 노드 메쉬형 네트워크에 비하여 처리율은 다소 떨어지고 블로킹 확률은 높아질 것으로 예측된다.

시뮬레이션을 위해 ns-2에서 제공되는 Static Routing을 이용하였으며 이는 Dijkstra의 SPF(Shortest Path First) 알고리즘을 이용한 것이며, 토폴로지에 대한 adjacency matrix와 링크의 비용을 사용하여 시뮬레이션이 시작되기 전에 경로가 계산되는 정적인 라우팅 방법이다.

트래픽의 발생은 패킷 스위칭을 위한 시뮬레이션에 적합하게 설계된 ns-2의 지수 분포에 따라 트래픽을 발생시키는 EXPOO_Traffic과 파레토 분포를 따르는 Poo_Traffic, 고정 비트율로 트래픽을 발생시키는 CBR_Traffic(Constant Bit Rate)을 사용하였다¹³⁾. 특히, 파레토 분포는 X축 변수값이 증가하여도 그 존재 확률이 지수 분포 보다 크게 나타나는 heavy-tail을 가지는 분포로서 음성 트래픽과는 속성이 다른 인터넷 트래픽

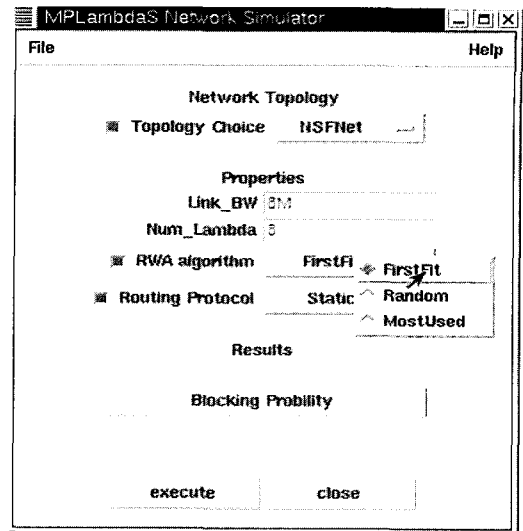


그림 9. 시뮬레이션 파라미터 입력 및 결과를 위한 사용자 인터페이스

Fig. 9. User interface for parameter input and results.

을 모델링 하는데 유용하다¹⁶⁾. 시뮬레이션 하는 동안 고정되어 있는 소스와 목적지 사이에만 세션을 생성하는 것은 충분치 않으므로 소스와 목적지는 랜덤하게 선택되어 모든 노드가 소스 혹은 목적지가 되게 하였다.

<그림 9>는 스크립트 기반의 시뮬레이션을 사용자가 용이하게 사용할 수 있도록 GUI 환경을 제공하는 것을 보여준다.

2. 시뮬레이션 결과

이 절에서는 위와 같은 환경의 MPλS 망에서 Static Routing 방법과 First-Fit 파장 할당 알고리즘을 사용한 시그널링과 패킷 포워딩 과정을 시뮬레이션하고, 트래픽의 부하와 사용되는 파장의 수를 배로 늘여가면서 MPλS 시뮬레이터의 단위 시간당 패킷의 처리율과 트래픽 부하에 따른 블로킹 확률을 구하여 결과를 보인다.

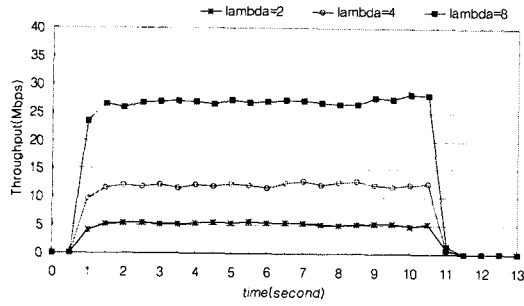
각 노드에서의 패킷의 파장 변환시 발생하는 시간지연은 고려하지 않았다. 또한 각 노드에서 패킷의 처리율과 연결요청된 광경로에 패킷을 할당하지 못했을 때의 블로킹 확률은 식 (2)와 식 (3)에 의해 구해진다.

$$\text{처리율} = \frac{\text{목적지에도착한패킷}(bit)}{\text{time}(s)} \times 10^6 \quad (2)$$

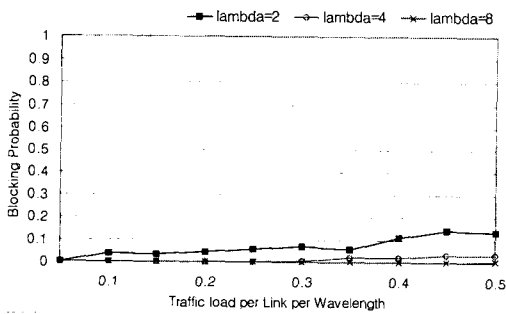
$$\text{블로킹 확률} = \frac{\text{블로킹된 광경로 개수}}{\text{요청된 총광경로 개수}} \quad (3)$$

<그림 10, 11> 및 <그림 12>는 앞에서 언급한 세 가지의 토폴로지에 대하여 시뮬레이션을 통해 출력된

XGraph 결과를 나타낸 것이다. NSFNet, 4x4 메쉬형 네트워크 및 30 노드 메쉬형 네트워크에서 트래픽 부하가 0.3일 때 파장의 수를 증가시키면서 시간에 따른 처리율과 트래픽 부하에 따른 블로킹 확률이다. 세 네트워크 토폴로지에 대하여 파장의 수가 늘어남에 따라 처리율이 높아지고 블로킹 확률이 커짐을 알 수 있다. 이러한

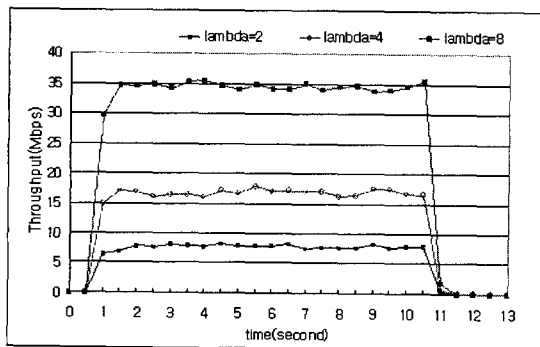


(a) 트래픽 부하 0.3에서의 처리율
(a) Throughput when network load is 0.3

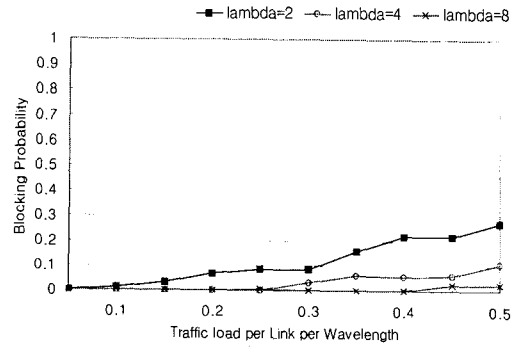


(b) 블로킹 확률
(b) Blocking Probability

그림 10. NSFNet에 대한 시뮬레이션 결과
Fig. 10. Simulation results for the NSFNet.

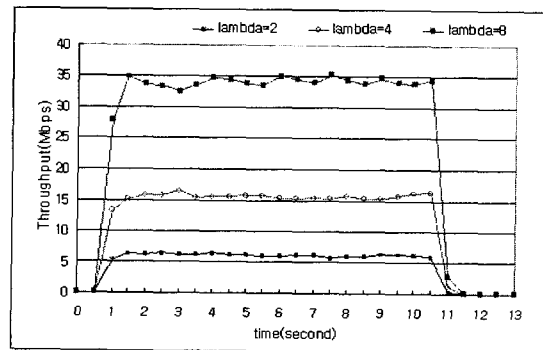


(a) 트래픽 부하 0.3에서의 처리율
(a) Throughput when network load is 0.3

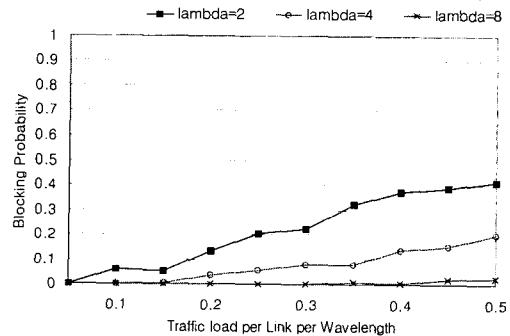


(b) 블로킹 확률
(b) Block Probability

그림 11. 4x4 메쉬 네트워크에 대한 시뮬레이션 결과
Fig. 11. Simulation results for the 4x4 mesh network.



(a) 트래픽 부하 0.3에서의 처리율
(a) Throughput when network load is 0.3



(b) 블로킹 확률
(b) Blocking Probability

그림 12. 30node mesh network에 대한 시뮬레이션 결과
Fig. 12. Simulation results for the 30 node mesh network.

결과는 패킷이 근원지 노드로부터 출발하여 목적지 노드로 이동함에 따라 중간 노드에서 할당 가능한 파장의

수가 줄어들게 되고 할당 가능한 파장이 없는 경우 블로킹이 발생하게 되기 때문이다. <그림 10>과 <그림 11>을 비교하면 동일한 트래픽 부하에서 메쉬형 네트워크의 처리율이 vBNS 네트워크 보다 높고 블로킹 확률도 보다 높게 나타남을 알 수 있다. 일반적으로 처리율이 높아지면 블로킹 확률이 낮아지지만 링크수의 증가에 따라 O-D 쌍의 증가율이 떨어짐으로 인한 결과로 판단된다. 이러한 시뮬레이션 결과는 IV장 1절에서 언급한 예측치와 일치하는 결과이다.

한편, <그림 11>과 <그림 12>를 비교하면 비슷한 형태의 세 개의 대표성을 갖는 네트워크에 대하여 파장의 수, 처리율, 블로킹 확률 등의 성능평가 파라미터를 이용하여 시뮬레이션 결과를 종합하면 시뮬레이션하기 이전의 예측 결과와 일치함을 알 수 있다. 이러한 결과로부터 본 논문에서 구현한 MPLS 시뮬레이터가 적절히 구현되었음을 알 수 있다. 따라서 본 시뮬레이터는 MPLS 프로토콜을 하드웨어로 구현하기 이전 단계에서 프로토콜의 기능 및 성능을 시험하고 검증하는데 유용하게 사용될 수 있을 것이다. 또한, 광네트워크를 설계하는 경우 주어진 네트워크 토폴로지에 대하여 필요한 링크의 수, 링크당 파장의 수 등을 결정하는데 중요한 역할을 할 수 있다.

V. 결 론

인터넷의 대중화 및 다양한 응용의 활성화에 힘입어 트래픽이 매우 빠르게 증가하고 있다. 이러한 트래픽을 수용하기 위하여 대용량의 대역폭을 제공하는 광인터넷 기술이 개발되고 있다. 현재 광인터넷 기술은 기존의 IP와 WDM을 통합한 구조를 기반으로 발전하고 있으며, 제어 및 시그널링 기능으로는 MPLS를 수정한 MPLS를 이용하고 있다. 이에 본 논문에서는 MPLS의 제어 기능들을 시험해보고 성능을 알아보기 위하여 MPLS 시뮬레이터를 설계하고 이를 ns-2 네트워크 시뮬레이터에 기반하여 기능을 구현하였다. 구현한 MPLS 시뮬레이터는 라우팅 정보에 기반하여 네트워크의 파장 정보를 각 노드에게 알려주는 시그널링 모듈과 파장 교환 기능을 하는 OXC 모듈, 파장변환기, 광경로의 설정과 파장 할당을 하는 RWA 모듈 등으로 구성되어 있다.

본 논문에서 구현한 MPLS 시뮬레이터의 성능을 평가하기 위하여 다양한 네트워크 토폴로지에 대하여 시뮬레이션을 통해 제공되는 파장의 수에 따른 처리율 및

트래픽 부하의 증가에 따른 블로킹 확률을 측정하였다. 이 결과들은 사전에 예측한 결과 값과 비교 분석하였다. 즉, 제공되는 파장이 많을수록 처리율이 증가하고, 트래픽의 부하가 증가할수록 블로킹 확률이 증가함을 확인할 수 있었으며 파장의 수가 늘어남에 따라 블로킹 확률은 감소함을 알 수 있었다. 이러한 결과는 네트워크의 토폴로지의 특성을 고려한 예측치와 일치하는 것이며 이러한 결과를 토대로 본 연구에서 구현한 시뮬레이터가 정상적으로 동작함을 알 수 있다. 이러한 시뮬레이터의 개발은 실제 광인터넷 구성 요소들을 개발하기 전 단계에서 WDM 및 MPLS의 기능을 검증하고 그 성능을 평가한다는 측면에서 매우 의미 있는 연구라 하겠다.

향후 연구 내용으로 대용량 대역폭을 제공하는 링크의 구현 및 파장을 더욱 효율적으로 할당하기 위한 알고리즘에 대한 연구가 필요할 것으로 사료된다. 또한, 망의 신뢰도를 높이기 위한 링크 에러 복구 및 링크 보호 기능에 대한 구현이 요구된다.

참 고 문 헌

- [1] "광인터넷 구축 방안에 관한 연구," 한국전산원, December, 2001년
- [2] D.O.Awduche, Y.Rekhter, J.Drake and R. Coltun, "Multi-Protocol Lambda Switching: Combining MPLS Traffic Engineering Control With Optical Crossconnects," Internet Draft, Work in Progress, Apr. 2001.
- [3] S. Seetharaman, A. Duresi, R. Jagannathan, R. Jain, N. Chandhok, K. Vinodkrishnan, "IP over Optical Networks: A Summary of Issues," Internet Draft draft-osu-ipo-mpls-issues-02.txt, April 2001.
- [4] K. Fall, "The ns Manual, The VINT Project," November 28, 2000.
- [5] Eric C. Rosen, Arun Viswanathan, Ross Callon, "Multiprotocol Label Switching Architecture," RFC 3031, Jan. 2001.
- [6] R. Callon et al., "A Framework for Multiprotocol Label Switching," Internet Draft, Sep. 1999.
- [7] Gae-IL Ahn, "Design and Implementation of MPLS Network Simulator," Ph.D. Dissertation, Chungnam Univ. Daejeon, Korea, Feb. 2002.

[8] 강민호, 이용규, 오세운, “광인터넷 기술발전 및 연구동향”, 대한전자공학회지, p.18-25, Vol.27, June, 2000.

[9] UCB/LBNL/VINT, “nsNoteandDocumentation,” <http://www-mash.cs.berkeley.edu/ns>.

[10] UCB/LBNL/VINT, “nsmanual,” <http://www-mash.cs.berkeley.edu/ns/ns-man.html>

[11] JaeChung, MarkClaypool, “NSbyExample,” <http://perform.wpi.edu/NS/index.html>.

[12] M Kova Cevic and A. S. Acampora, “Benefits of wavelength translation in all-optical clear channel networks,” IEEE Journal on Selected Areas in Comm., vol. 14, no. 5, pp.780-799, June 1996.

[13] 이쌍수, 이창형, 서승우, 장문중, “파장 분할 다중화(WDM) 망의 분산 파장 할당 알고리즘,” 한국통신학회논문지, Vol. 24, No. 2A, 1999

[14] E. Karasan and E. Ayanoglu, “Effects of Wavelength Routing and Selection Algorithms on Wavelength Conversion Gain in WDM Networks,” IEEE Transactions on Networking, Vol. 6, pp. 186-196, April 1998.

[15] Bo wen, Niles M. Bhide, Ramakrishna K. Shenai, Krishna M. Sivalingam, “Optical Wavelength Division Multiplexing (WDM) Network Simulator(OWNS): Architecture and Performances Studies,” SPIE Optical Networks Magazine Special Issue on Simulation, CAD, and Measurement of Optical Networks, March 2001.

[16] “DQDB 프로토콜의 성능분석을 위한 컴퓨터 시뮬레이션,” http://cnr.kaist.ac.kr/lecture/ee520_2002/project1/Project1.doc

저 자 소 개



徐 善 英(正會員)
 2001년 : 대전대학교 정보통신공학과 졸업(공학사). 2003년 : 대전대학교 대학원 정보통신공학과 졸업(공학석사). <주관심분야 : 광인터넷, 그리드 프로토콜, 네트워크 보안>



黃 善 奘(正會員)
 1979년 2월 : 서강대학교 수학과 졸업(이학사). 1987년 1월 : Case Western Reserve University 전자계산학과 졸업(공학석사). 1993년 5월 : Case Western Reserve University 전자계산학과 졸업(공학박사). 1979년~1982년 : 한국과학기술연구원 근무. 1993년~1995년 : (주)현대전자연구소 근무. 1995년~현재 : 대전대학교 정보통신인터넷공학부 부교수. <주관심분야 : Smart Card, Security, VLSI Testing>



李 奉 煥(正會員)
 1985년 : 서강대학교 전자공학과 졸업(공학사). 1987년 : 연세대학교 대학원 전자공학과 졸업(공학석사). 1993년 : Texas A&M 대학교 전기공학과 졸업(공학박사). 1987년~1995년 : 한국통신 연구개발원 연구원. 1995년~현재 : 대전대학교 정보통신공학과 부교수. <주관심분야 : 광인터넷, 그리드 프로토콜, 네트워크 보안>



尹 贊 鉉(正會員)
 1981년 : 경북대학교 전자공학과 졸업(공학사). 1985년 : 경북대학교 대학원 전자공학과 졸업(공학석사). 1994년 : 일본 동북대학교 전기통신공학과 졸업(공학박사). 1986년~1997년 : 한국통신 통신망연구소 실장. 1997년~현재 : 한국정보통신대학원대학교 공학부 부교수. <주관심분야 : 광인터넷, 그리드미들웨어, 네트워크 성능분석>