

論文2003-40SD-7-7

메쉬 기반의 생존성 WDM 망의 설계 기법에 관한 연구

(A Study on Design Scheme of Mesh-Based Survivable WDM Networks)

玄基豪*, 鄭榮哲*

(Ki-Ho Hyun and Young-Chul Chung)

요약

메쉬 기반의 파장분할다중화(WDM)망에서는 단 하나의 광섬유 케이블의 장애로도 심각한 데이터의 손실을 야기할 수 있다. 따라서, 이러한 장애에 대한 효율적인 복구 알고리즘이 필요하다. 본 논문에서는 메쉬 기반의 WDM 망에서의 복구 기법에 대한 새로운 알고리즘을 제안하였고 기존의 복구 기법과 비교하였다. 복구 방법으로는 경로 복구, 링크 복구의 대표적인 두 가지 기법이 있다. 이러한 기법들의 한 가지 공통점은 단일 링크장애에 대해서 논하고 있다는 것이다. 본 논문에서 제안한 복구 기법에서는 각각의 링크 장애에 대한 복구 경로를 확보하지 않는다. 메쉬 기반의 WDM망을 여러 개의 작은 페루프로 분리하고 페루프 내에서 단일 링크 장애가 발생하면 그 페루프를 우회하여 복구 경로가 결정되는 복구 기법과 복구 경로가 페루프 내에서만 결정되는 복구 기법을 제안하였다. 경로 복구 기법과 링크 복구 기법과 더불어 제안된 기법의 성능을 비교하였다. 시뮬레이션 결과 제안된 두 기법은 비록 총 사용되는 파장 마일리지가 기존의 복구 기법보다 10%에서 50% 정도 증가하게 되지만 경로 설정에 소요되는 실행시간(CPU Time)은 경로 복구 기법과 링크 복구 기법에 비해 상당히 감소되는 결과를 얻게 되었다.

Abstract

A single fiber failure in mesh-based WDM networks may result in the loss of a large number of data. To remedy this problem, an efficient restoration algorithm for a single fiber failure in the mesh-based WDM network is necessary. We propose a new algorithm for restoration scheme in WDM networks and compare it with previous schemes. Path restoration and link restoration are two representative restoration schemes which deal with only a single link failure. In this paper, we propose two kinds of efficient restoration scheme. In the proposed schemes the restoration path for each link failure is not secured. The mesh network is decomposed into a number of small loops. In one algorithm, any link failure in a certain loop is regarded as the failure of the loop and the restoration lightpath is selected by detouring the failed loop. In another scheme any link failure in a certain loop is restored within the loop. We compare performance of the proposed schemes with conventional path restoration scheme and link restoration scheme. Simulation results show that CPU time in the proposed schemes decreases compared with that in path restoration scheme and link restoration scheme, although total wavelength mileage usage increases by 10% to 50%.

Keyword : Optical network, survivable network, restoration, wavelength division multiplexing, (WDM), integer linear programming.

* 正會員, 光云大學校 電子工學部/電子通信工學科
(School of Electronics Engineering, Kwangwoon

University)

接受日:2002年2月21日, 수정완료일:2003년7월4日

I. 서론

최근 인터넷 서비스에 대한 수요가 폭발적으로 증가함에 따라서 현재의 네트워크가 제공하는 대역폭으로는 대용량의 정보를 원활하게 수용하기에는 어려움이 있다. 다양한 대용량의 멀티미디어 트래픽을 안정적이고 고품질의 통신 서비스로 전달하기 위한 유력한 방안으로써 파장분할 방식(WDM: Wavelength Division Multiplexing)의 전광네트워크(all-optical network)가 전 세계적으로 확산되고 있다^[1].

파장 분할 다중화(WDM) 기술은 광섬유의 넓은 주파수 대역을 파장 분할하여 각 파장 별로 수 Gbit/s에서 수십 Gbit/s의 데이터를 전송함으로써 하나의 광섬유를 통해 Tbit/s급의 데이터를 전송할 수 있는 기술이다. 이렇게 한 링크를 통해 전송되는 트래픽의 크기가 증가함에 따라 WDM 망에서는 단일 선로의 장애도 심각한 데이터 손실을 초래한다. 따라서 이러한 장애에 대처할 수 있도록 설계된 망이 필요하게 되는데 이러한 망을 생존성 망(survivable network)이라고 한다.

일반적인 WDM망에서의 장애 복구는 두 가지 방식으로 나눌 수 있는데, 하나는 보호(Protection)이고 다른 하나는 복구(Restoration)이다. 보호는 운용경로에 대해 미리 보호경로를 사전계획(preplanning)해 두었다가 고장이 발생하면 지정된 보호 경로를 따라 운용경로를 우회하는 방식이다. 보호의 장점은 빠르고 간단하지만 단점으로는 예비용량이 많이 필요하다는 것이 단점이 된다. 보호의 대표적인 예로는 1+1 보호, N:M 보호, 1:1 보호, Sonet BLSR, UPSR, MPLS TE 경로 보호가 있다. 복구는 보호 경로를 미리 지정해 두지 않고 고장이 발생한 시점의 망 상태에 따라 남아있는 예비용량을 이용하여 보호 경로를 실시간으로 확보하여 장애가 발생한 트래픽을 우회하는 방식이다. 복구의 장점은 예비용량이 보호에 비해 적게 사용이 되고 단점으로는 보호에 비해 느리며 복잡하다. 복구의 대표적인 예로는 IP routing protocol, GMPLS path protection이 있다.

<그림 1>에 보호/복구의 분류를 나타내었다. 전용보호는 하나의 운용경로에 대해 배타적으로 하나의 예비 경로를 지정해 놓는 방식이며 공유보호는 여러 개의 운용 경로가 예비 경로를 공유하는 방식이다. 일반적으로 서로 다른 링크에서의 장애 발생시 사용되는 예비

경로를 공유할 수 있도록 설계를 할 수 있다.

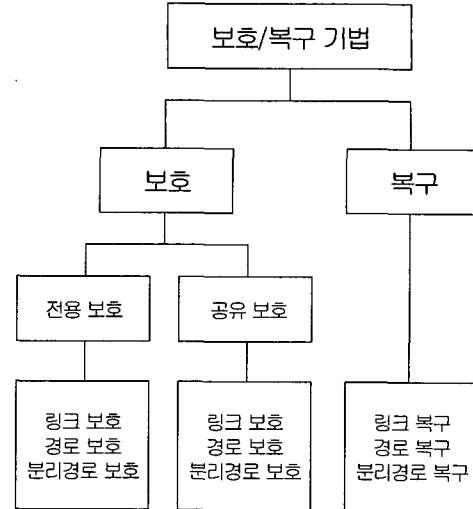


그림 1. 보호/복구의 분류

Fig. 1. Classification of Protection/Restoration.

메쉬 기반의 WDM망에서의 장애 복구는 크게 두 가지 방식으로 나눌 수 있는데, 하나는 사전 계획을 통한 예비 용량 설정 방식(Pre-planned spare capacity assignment)이고 다른 하나는 실시간 우회 방식이다. 실시간 우회 방식은 트래픽의 동적 특성을 수용할 수 있는 장점이 있지만, 장애 발생시의 망 상태에 따라 트래픽 복구를 완전히 보장할 수 없는 경우가 발생할 수 있다는 단점이 있다^[2]. 따라서 실시간 우회 방식은 특정 계층에서 트래픽을 완전히 복구하지 않아도 상위 계층의 복구 기능과 연동 가능한 경우에 적용될 수 있다. 이에 대한 대표적인 예로는 IP over WDM과 같이 광 계층의 복구 기능이 상호 보완적인 경우를 들 수 있다. 하지만 이 경우 광 계층의 복구 기능과 상위 계층의 복구 기능을 통합하여 제어하는 기술이 아직 초기 논의 단계에 머물고 있는 실정이다^[3, 4].

사전 예비 용량 설정 방식은 실시간 우회 방식과 달리 트래픽을 정적으로 가정하고 예비 용량을 미리 적절하게 배치함으로써 장애 발생 시에 생존할 수 있는 망을 설계하는 방식이다. 이에 대한 연구는 과거 DCS(Digital Crossconnects)를 사용한 메쉬 망이나 ATM VP (Virtual Path)망 상에서 많은 연구가 이루어졌다^[6-12].

사전 예비 용량 설정 문제를 풀기 위한 대표적인 두 가지 기법으로는 링크 복구(LR: Link Restoration)와

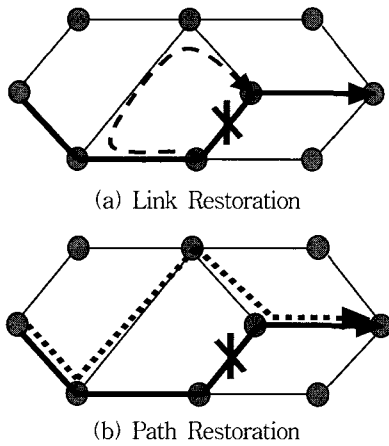


그림 2. 대표적인 복구 기법.
Fig. 2. Representative restoration schemes.

경로 복구(PR: Path Restoration)이다^[3,5]. <그림 2(a)>는 링크 복구를 나타낸 것으로 굵은 실선은 운용 경로를 나타내고 X 표시가 된 곳은 링크가 절단된 것을 나타낸다. 링크보호의 경우, 장애 발생 시 절단된 링크의 양 끝점 노드 사이에 점선과 같은 예비 경로를 확보함으로써 절단된 링크를 복구하는 기법이다. 이 기법은 설계 알고리즘의 복잡도가 낮고 복구에 걸리는 시간이 짧으나 경로 보호에 비해 많은 보호 용량을 필요로 한다. 반면, <그림 2(b)>의 경로 복구의 경우 절단된 링크를 지나는 운용 경로 전체에 대해 트래픽의 시작노드부터 도착노드까지를 새로운 복구 경로로 연결하게 된다. 경로 복구 기법은 링크 복구보다 훨씬 적은 용량으로 원하는 수준의 복구 기능을 보장하고 하지만 알고리즘의 복잡도가 링크 복구에 비해 상당히 높고 고장이 났을 경우 트래픽의 시작 노드까지 장애 발생 신호가 전달된 후에야 트래픽을 우회할 수 있기 때문에 복구 시간이 길어지게 된다. <그림 2(b)>에서는 하나의 광 경로에 대해 복구 경로만을 나타냈지만, 실제 WDM망에서는 한 링크를 지나는 운용경로가 수십 개에 해당하기 때문에 링크가 절단된 경우 그 영향을 받은 모든 연결 노드 쌍에 대해 그림과 같이 보호 경로를 확보해 주어야 한다.

본 연구에서는 일반적인 단일 링크 장애에 대한 복구 경로 확보를 하지 않고 전체 네트워크를 몇 개의 폐루프(closed loop)로 분할하여 그 폐루프 내에서 장애 발생시 그 루프를 우회하여 복구 경로가 확보되어지는 알고리즘을 제안하였다. 그리고 경로 복구 기법과 비교하여 그 폐루프에서 어느 한 링크가 장애가 발생하였

을 경우 그 폐루프 내에서 링크 복구 메커니즘을 이용하여 복구 경로가 확보되어지는 간단한 알고리즘을 제안하였다.

이러한 두 가지 새로운 알고리즘을 사용하게 되면 Loop 링크 복구(Loop LR)기법과 Loop 경로 복구(Loop PR)기법인 경우에 두 기법 모두 자원 요구량은 기존의 기법들 보다 증가하게 되지만 정수 선형 계획법을 이용하여 최적의 해를 구하는 시간은 기존의 기법들보다는 상당한 시간이 감소된다는 것을 알 수 있다.

논문을 다음과 같이 구성된다. 2장에서는 기존 기법의 정수 선형 계획법에 대해서 분석하고 3장에서는 제안하는 알고리즘과 제안하는 기법의 정수 선형 계획법을 제시한다. 4장에서는 기존 기법과 제안한 기법의 시뮬레이션 결과와 분석을 기술하며, 마지막으로 5장에서 본 연구의 결론을 맺는다.

II. 정수 선형 계획법

이 장에서는 메쉬 기반의 WDM망에서의 대표적인 복구 기법들에 대한 복구 시간(Restoration Time)과 각각의 정수 선형 계획법을 알아보도록 한다.

1. 복구 시간

이 절에서는 LR와 PR의 복구 시간을 분석한다. 복구 절차는 [12]에 제시된 모델을 사용한다. LR의 복구 절차는 다음 순서에 따라 이루어진다.

- ① 장애가 발생한 링크의 앞쪽 노드에서 장애를 감지한다. 이 때 장애를 감지하는데 걸리는 시간은 F로 정의한다.
- ② 장애를 감지한 노드에서 각각의 장애가 발생한 연결들의 목적지 노드로 미리 정의된 복구 경로를 따라 복구 경로 설정 메시지를 보낸다. D를 복구 메시지 처리 시간, P를 링크에서 메시지가 전파되는데 걸리는 시간, m_l 을 LR의 복구 경로의 평균 홉 길이라고 정의하면 이 과정에서 걸리는 시간은 $(m_l + 1)D + m_l P$ 이 된다.
- ③ 각 목적지 노드는 복구 경로 메시지에 대한 확인 메시지를 복구 경로를 통해 보낸다. 확인 메시지가 복구 경로를 따라 전달되는 과정에서 복구 경로를 예약하게 되며 복구 경로 상의 각 노드들은 광교환기를 복구 경로에 맞게 설정한다. 이 과정의 소유시간은 $(m_l + 1)D + m_l P + (m_l + 1)C$ 이며, 여기서 C는 광

교환기 스위칭 시간을 나타낸다.

- ④ 노드에 확인 메시지가 도착하면 장애가 발생한 연결을 새로운 복구 경로로 우회시킨다.
LR기법의 복구 시간은 1-4에서 걸린 시간의 합으로 $F+(m+1)C+2(m+1)d+2mP$ 이 된다. 같은 방법으로 PR에 대해 복구 시간을 구해보면 $F+n_p+(n+1)D+(m_p+1)C+2(m_p+1)d+2m_pP$ 을 얻을 수 있으며(n 은 장애가 발생한 링크의 앞쪽 노드에서 연결의 발생점 노드까지의 홉 길이, m_p 는 PR의 복구 경로의 평균 홉 길이) 여기서 추가된 $(n+1)D$ 항은 장애가 발생한 링크에서 트래픽의 발생점으로 경보 신호가 전달되는데 걸리는 시간이다. 위의 두 가지 수식을 비교하여 보면 PR의 복구 시간이 LR보다 길다는 것을 확인할 수 있는데, 그 이유는 PR에서 알람신호가 고장난 링크에서 발생점 노드로 전달되는 시간이 추가적으로 필요하기 때문이다. 또한 복구 경로의 길이에 있어서 PR는 복구 경로가 트래픽의 양 끝점 사이에 형성된다는 점에서 LR에 비해 상대적으로 복구 경로가 길어질 것을 예상할 수 있다.

2. LR 기법의 정수 선형 계획법

2.1 변수 정의

공통변수 :

- $G(V, E)$: 방향성그래프 V 는 망 상의 모든 노드의 집합, E 는 망 상의 모든 링크의 집합.
- $N=|V|, L=|E|$.
- A_{sd} : 노드 쌍 (s, d) 에서의 총 연결요청의 개수.
- D : 거리 행렬, $d_{mn} = \infty$ 이면 링크 $(m, n) \notin E$ 이다.
- M_{sd} : 노드 쌍 (s, d) 사이 가능한 운용 경로 후보의 집합. 운용 경로들은 1에서 j 번째 순서로 되어있다.
- T_{sd}^j : 노드 쌍 (s, d) 를 갖는 후보 집합에서 j 번째 운용 경로
- P_a : a 링크에서의 복구 경로의 총 개수.
- $P_{a,b}$: b 링크에서 장애 발생시 a 링크에서의 복구 경로 개수.
- C_{sd}^j : 노드 쌍 (s, d) 를 갖는 집합중 j 번째 운용 경로의 총 길이
- C_a : a 링크의 거리

LR 기법에 사용되는 변수들 :

- LRP_b : b 링크에 장애가 발생했을 경우 링크 b 의 양 끝 노드를 연결하는데 사용되는 예비 경로 후보들의 집합. 예비 복구 경로들은 1에서 q 번째 순서로 되어 있다.
- β_a^q : a 링크에 장애가 발생했을 경우 q 번째 예비 경로 후보가 사용되고 있는가를 나타내는 이진수.
- $\theta_b^{q,a}$: b 링크에 장애가 발생했을 경우 q 번째 예비 경로 후보 상에 a 링크가 포함되는가를 나타내는 이진수
- $\delta_{sd,a}^j$: 노드 쌍 (s, d) 간의 j 번째 운용 경로 후보에 a 링크가 포함되는가를 나타내는 이진수.

2.2 정수 선형 계획법

목적함수 :

$$\text{Min} : \sum_{sd} \sum_{j \in M_{sd}} C_{sd}^j T_{sd}^j + \sum_a C_a P_a \quad (2.1)$$

식 (2.1)은 총 운용경로의 마일리지와 총 복구 경로의 마일리지의 합을 최소화시키는 목적함수이다.

제약조건 :

$$\sum_{j \in M_{sd}} T_{sd}^j = A_{sd} \quad \forall (s, d) \in V \quad (2.2)$$

$$\sum_{j \in M_{sd}} T_{sd}^j \cdot \delta_{sd,a}^j = \sum_{q \in LRP} \beta_a^q \quad (2.3)$$

$$\forall (s, d) \in V \quad \forall b \in L \quad \forall a \in L \quad (2.3)$$

$$P_{a,b} = \sum_{q \in LRP} \beta_b^q \cdot \theta_b^{q,a} \quad \forall b \in L \quad \forall a \in L \quad a \neq b \quad (2.4)$$

$$P_a = \max_{b \in L} P_{a,b} \quad \forall b \in L \quad \forall a \in L \quad a \neq b \quad (2.5)$$

3. PR 기법의 정수 선형 계획법

3.1 변수 정의

PR 기법에 사용되는 변수들 :

- PRP_b^{sd} : b 링크에 장애가 발생시 노드 쌍 (s, d) 간의 가능한 예비 복구 경로들의 후보 집합. 예비 복구 경로들은 1에서 q 번째 순서로 되어 있다.
- $\beta_{sd,a}^q$: b 링크에 장애가 발생했을 경우 노드 쌍 (s, d) 간의 q 번째 예비 경로 후보가 사용되고 있는가를 나타내는 이진수.

- $\theta_{sd,a}^a$: b 링크에 장애가 발생했을 경우 노드 쌍 (s, d) 간의 q번째 예비 경로 후보 상에 a 링크가 포함되는가를 나타내는 이진수
- $\delta_{sd,a}^j$: 노드 쌍 (s, d) 간의 j번째 운용 경로 후보에 a링크가 포함되는가를 나타내는 이진수.

3.2. 정수 선형 계획법

목적함수 :

$$Min : \sum_{sd} \sum_{j \in M_{sd}} C_{sd}^j T_{sd}^j + \sum_a C_a P_a \quad (3.1)$$

식 (3.1)은 총 운용경로의 마일리지와 총 복구 경로의 마일리지의 합을 최소화시키는 목적함수이다.

제약조건 :

$$\sum_{j \in M_{sd}} T_{sd}^j = \Lambda_{sd} \quad \forall (s, d) \in V \quad (3.2)$$

$$\sum_{j \in M_{sd}} T_{sd}^j \cdot \delta_{sd,a}^j = \sum_{q \in PRP} \beta_{sd,a}^q \quad (3.3)$$

$$\forall (s, d) \in V \quad \forall a \in L$$

$$P_{a,b} = \sum_{sd} \sum_{q \in PRP} \beta_{sd,b}^q \cdot \theta_{sd,a}^a \quad (3.4)$$

$$\forall b \in L \quad \forall a \in L \quad a \neq b$$

$$P_a = \max_{b \in L} P_{a,b} \quad \forall b \in L \quad \forall a \in L \quad a \neq b \quad (3.5)$$

식 (2.2)와 식 (3.2)는 연결 설정 요청이 전부 만족되어야 하는 제약조건이며 식 (2.3)와 식 (3.3)은 장애 링크 a가 발생하였을 경우 그 장애링크 a를 지나는 트래픽은 복구 경로를 통해 완전히 복구 되어져야 하는 제약조건이다. 식 (2.4)와 식 (3.4)는 링크 b에 장애가 발생하였을 경우 링크 a에서 필요한 복구 경로의 개수를 구하는 제약조건이다. 식 (2.5)와 식 (3.5)는 링크 b (b≠a)에 장애가 발생하였을 경우 링크 a에서 필요한 복구 경로의 개수 $P_{a,b}$ 중 최대값을 가져야 하는 제약조건이다.

III. 제안한 알고리즘

본 절에서 최적의 해를 빠른 시간내에 구하기 위해서 두 개의 새로운 알고리즘을 구현하였다. 다음은 두 알고리즘에서 고려한 사항과 수행절차 그리고 정수 선형 계획법에 대해서 설명한다.

1. Loop LR 알고리즘

1.1. 알고리즘

이 알고리즘은 상당히 간단한 메카니즘으로 구성되어 있으며, LR 기법과 유사하지만 Loop LR기법은 페루프 내에서 링크에 장애가 발생이 되면 그 루프내에서 복구 경로가 확보되지만 LR기법은 단지 링크를 우회하는 복구경로를 확보하는 것이기 때문에 제안한 알고리즘과 차이가 있다. <그림 3>은 Loop LR 알고리즘을 구현하는 순서도이다. 정수 선형 계획법을 적용하기 전까지 페루프를 탐색하여 전체 망의 모든 링크를 만족할 수 있는 페루프를 선택하고 선택된 페루프들 사이에서 공유되는 링크를 어느 하나의 페루프로 포함시켜 정수 선형 계획법으로 최적의 해를 구하는 것이다.

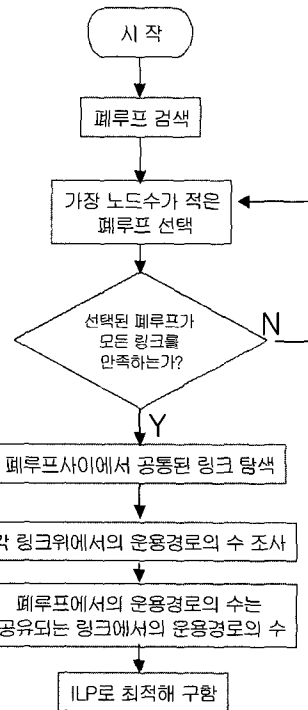


그림 3. Loop LR 알고리즘 구현 순서도
Fig. 3. Flow chart for Loop LR algorithm.

<그림 4(a)>와 <그림 4(b)>는 각각의 다른 경로를 가지고 있는 경우에 대해서 동일 페루프내에서 장애 복구에 대한 예시를 보여주고 있다. <그림 4(a), (b)>는 선택된 페루프 사이에서 어느 한 링크에 장애가 발생되었을 경우 그 페루프 안에서 복구 경로가 구성되지 않고 공유되는 링크는 각 페루프의 트래픽의 양에

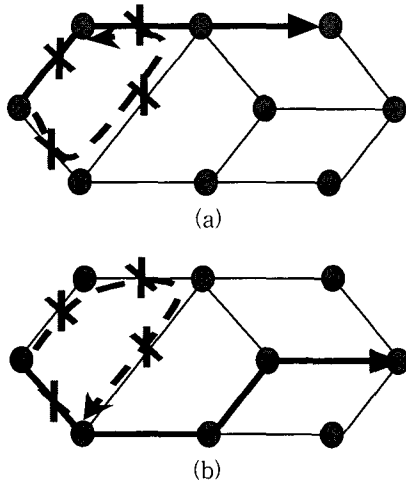


그림 4. Loop LR 기법
Fig. 4. Loop LR scheme.

따라서 트래픽의 양이 많은 페루프로 복구 경로가 구성된다. 이 페루프만 고려할 경우에는 Loop LR의 복구 시간은 LR기법과 동일하다.

1.2. Loop LR에 대한 정수 선형 계획법 변수 정의 :

- $kc1_a^A$: 페루프 A에 링크 a가 포함되는가를 나타내는 이진수
- $kc2_a^A$: 페루프 A의 반대방향으로 링크 a가 포함되는가를 나타내는 이진수
- $p1^A$: 페루프 A에서 필요한 최대 복구 경로의 수.
- $p2^A$: 페루프 A의 반대 방향으로 필요한 최대 복구 경로의 수
- $kd_{sd,j}^a$: 노드 쌍 (s, d) 간의 j번째 경로에 링크 a가 포함되는가를 나타내는 이진수

정수 선형 계획법 :
목적함수 :

$$Min : \sum_{sd} \sum_{j \in M_{sd}} C_{sd}^j T_{sd}^j + \sum_a C_a P_a \quad (1.1)$$

식 (1.1)은 총 운용경로의 마일리지와 총 복구 경로의 마일리지의 합을 최소화시키는 목적함수이다.

제약조건 :

$$\sum_{sd} \sum_{j \in M_{sd}} T_{sd}^j = \Lambda_{sd} \quad \forall (s, d) \in V \quad (1.2)$$

$$\sum_A (kc1_a^A \cdot p1^A + kc2_a^A \cdot p2^A) \geq \sum_{sd} \sum_{j \in M_{sd}} kd_{sd,j}^a \cdot T_{sd}^j \quad (1.3)$$

$$P_a = \max_{A \in LA} \{kc1_a^A \cdot p1^A, kc2_a^A \cdot p2^A\} \quad (1.4)$$

$$\forall a \in L \quad \forall A \in CL$$

식 (1.2)는 연결 설정 요청이 전부 만족되어야 하는 제약조건이며, 식 (1.3)은 페루프로 구성된 링크를 지나 는 운용경로의 수를 구하는 제약조건이다. 식 (1.4)은 페루프내의 각 링크에서 운용경로의 최대의 경로 수를 구하여 페루프의 복구 경로의 수를 구하는 제약조건이다.

2. Loop PR 알고리즘

2.1. 알고리즘

기존의 알고리즘은 단일 링크 장애 발생에 대한 복구 경로를 확보하는 기법이지만 본 논문에서 제안하는 알고리즘은 페루프의 어느 한 링크에 장애가 발생하였을 경우 페루프 자체를 우회하여 복구 경로를 확보하는 기법이다. 만약 트래픽의 시작과 끝점이 페루프 내에서만 존재하는 경우에는 그 트래픽에 대해서는 장애 발생 페루프 내에서 복구 경로가 확보되어야 한다.

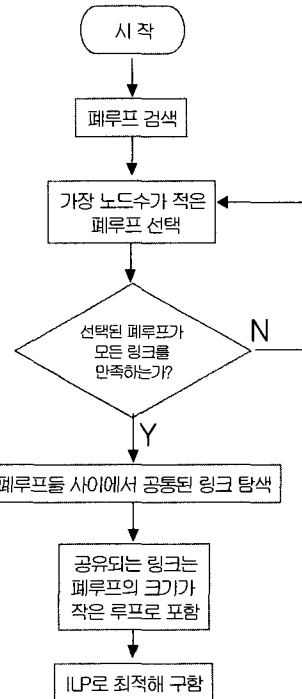


그림 5. Loop PR 알고리즘 구현 순서도
Fig. 5. Flow chart of Loop PR algorithm.

<그림 5>는 Loop PR 알고리즘을 구현하는 순서도이다. 정수 선형 계획법을 적용하기 전까지 페루프를 탐색하여 전체 망의 모든 링크를 만족할 수 있는 페루프 및 가장 적은 노드수를 갖는 페루프를 선택하며 공유 링크에 대해서는 어느 한 페루프에 포함시켜야 하는데 그 이유는 페루프 장애가 발생하였을 경우 공유 링크가 중복되기 때문에 전체 마일리지가 증가하게 된다. 따라서 전체 마일리지를 감소시키기 위해서는 공유 링크를 어느 하나의 페루프로 포함시키고 다른 페루프는 그 공유링크를 포함하지 않는 페루프로 선택을 하게 된다. 따라서 공유링크를 어느 하나의 페루프로 포함시킨 후 정수 선형 계획법으로 최적의 해를 구하는 것이다.

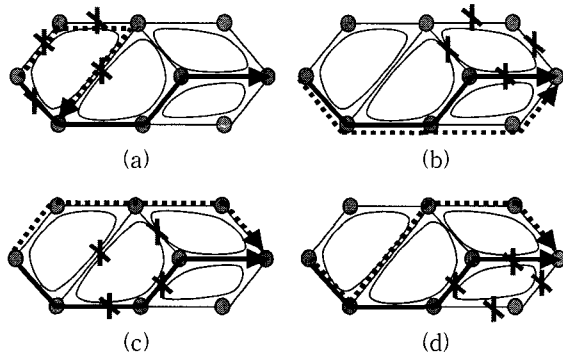


그림 6. Loop PR 기법
Fig. 6. Loop PR scheme.

<그림 6>는 제시된 네트워크 모형에서 어느 한 트래픽에 대해서 복구 경로가 이루어지는 과정을 보여주고 있다. 이 모형에서 4개의 페루프로 구성되어 있는데 각각의 페루프 장애시 복구 경로가 어떻게 이루어지는지 보여주고 있다. <그림 6(a)>의 경우에는 복구 경로가 이루어지지 않지만 <그림 6(b), (c), (d)>는 페루프의 어느 한 링크에서 장애가 발생하였을 경우에 복구되는 경로가 존재를 하게 된다. 이런 경우에 대비하여 이러한 트래픽에 대해서는 Loop LR 알고리즘을 이용해 그 자체내 페루프를 통해서 이루어지게 된다. Loop PR의 복구 시간은 PR기법의 복구 시간과 동일하게 된다.

2.2. Loop PR에 대한 정수 선형 계획법
변수 정의 :

- LPR_A : 페루프 A에 장애가 발생했을 경우 페루프 A를 우회하는 복구 경로의 집합.

- $\delta_{sd,A}^j$: 노드 쌍 (s, d) 간의 j 번째 경로에 페루프 A를 이루고 있는 링크가 포함되는가를 나타내는 이진수
- $\beta_{sd,A}^q$: 노드 쌍 (s, d) 간의 페루프 A를 포함하지 않는 q 번째 복구 경로
- $P_{a,A}$: 페루프 A에 장애가 발생하였을 경우 링크 a 에서 필요한 복구 경로의 개수.
- $\theta_{sd,a}^j$: 노드 쌍 (s, d) 간의 q 번째 복구 경로 후보에 a 링크가 포함되는가를 나타내는 이진수.

정수 선형 계획법
목적함수 :

$$Min : \sum_{sd} \sum_{j \in M_{sd}} C_{sd}^j T_{sd}^j + \sum_a C_a P_a \quad (2.1)$$

식 (2.1)은 총 운용경로의 마일리지와 총 복구 경로의 마일리지의 합을 최소화시키는 목적함수이다.

제약조건 :

$$\sum_{sd} \sum_{j \in M_{sd}} T_{sd}^j = \Lambda_{sd} \quad \forall (s, d) \in V \quad (2.2)$$

$$\sum_{j \in M_{sd}} T_{sd}^j \cdot \delta_{sd,A}^j = \sum_{q \in LPR_A} \beta_{sd,A}^q \quad \forall (s, d) \in V \quad \forall A \in CL \quad (2.3)$$

$$\sum_A (kc1_a^A \cdot p1^A + kc2_a^A \cdot p2^A) \geq \sum_{sd} \sum_{j \in M_{sd}} \delta_{sd,a}^j \cdot T_{sd}^j \quad \forall (s, d) \in V \quad s \neq d \quad \forall a \in L \quad (2.4)$$

$$R_a \geq kc1_a^A \cdot p1^A, \quad R_a \geq kc2_a^A \cdot p2^A \quad \forall a \in L \quad \forall A \in CL \quad (2.5)$$

$$P_{a,A} = \sum_{q \in LPR_A} \beta_{sd,A}^q \cdot \theta_{sd,a}^j \quad \forall (s, d) \in V \quad \forall A \in CL \quad (2.6)$$

$$P_a = \max_{A \in CL} P_{a,A} + R_a \quad \forall A \in CL \quad \forall a \in L \quad (2.7)$$

식 (2.2)는 연결 설정 요청이 전부 만족되어야 하는 제약조건이며, 식 (2.3)은 페루프의 어느 한 링크에 장애가 발생시 그 페루프를 우회하는 복구 경로를 구하는 제약조건이다. 식 (2.3)과 식 (2.4), 식 (2.5)는 페루프를 우회하는 복구 경로가 없을 경우에 사용되는 제약조건이다. 이 세 개의 식은 Loop LR로 구성되어져 있다. 식 (2.6)은 페루프 A에 장애가 발생하였을 경우에 링크 a 에서 필요한 복구 경로의 개수를 구하는 제

약조건이다. 식 (2.7)은 페루프를 우회하는 복구 경로 ($P_{a,A}$)와 페루프의 각 링크에서 가지는 복구경로(R_a)의 합의 최대값을 구하는 제약조건이다.

IV. 시뮬레이션 및 결과

PR, LR과 두 개의 제안한 알고리즘으로 시뮬레이션을 8-노드 망과 14-노드 NSFNET에 대하여 수행하였다. 각각의 시뮬레이션에 사용된 망에 대한 그림은 <그림 7>와 <그림 8>에 나타나 있다. <그림 7>에서 각 링크 위의 숫자의 단위는 km로 실제 거리를 근사화한 것이다. <그림 8>은 [12]에서 사용된 것으로 각 링크 위의 숫자는 상대적 링크 가중치를 나타낸다. 노드 간 연결 링크는 양방향 링크로 가정하는데 이 것은 두 개의 단 방향 링크가 서로 반대 방향으로 놓여져 있는 상태를 의미한다. 연결 요청 행렬은 <표 1>에 나타나 있으며 이는 8-노드 망의 성능을 평가하는데 쓰인다. 다른 두 개의 연결 설정 요청 행렬인 <표 2>와 <표 3>은 [12]에 나타난 트래픽 행렬을 10으로 나눈 후 정수 값을 취한 결과로 NSFNET의 성능평가에 쓰인다. 연결 요청 행렬 T2는 35개의 연결 요청 쌍을 가지고

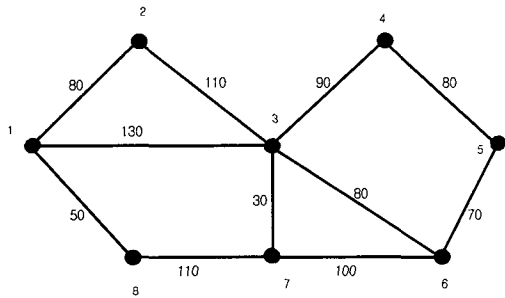


그림 7. 8-노드 망
Fig. 7. 8-node network.

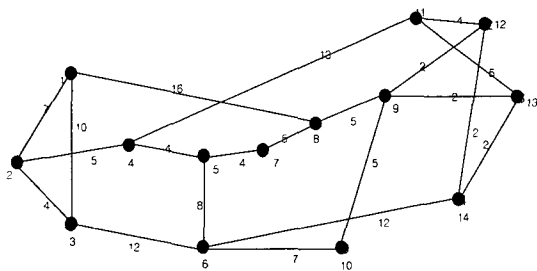


그림 8. 14-노드 NSFNET
Fig. 8. 14-node NSFNET.

표 1. 8-노드 망의 연결요청 행렬 T1
Table 1. T1 traffic demand matrix of 8-node network.

노드	1	2	3	4	5	6	7	8
1		1	1	4	5	4	2	8
2	1				1			1
3	1				1			1
4	3				2	2	1	3
5	5	1	1	2		2	1	4
6	4	1		2	2		1	3
7	2			1	1	1		2
8	8	1	1	4	5	4	2	

표 2. 14-노드 NSFNET의 연결요청 행렬 T2
Table 2. Traffic demand matrix T2 for 14-node NSFNET.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1		3	3	3			8	3		4				
2								6						
3	4						3		6		2			
4								9						
5		10		5										
6														9
7										7	6			7
8				10										8
9	1						2							
10					5	1		5				6		
11								3					7	
12								4		10				
13	2				4	5								
14	1				8									

있고 연결 요청 행렬 T3는 79개의 연결 요청 쌍을 가지고 있다. 두 개의 네트워크 모델에서 링크의 파장 제한은 없으며 사용 패키지는 공개용 ILP(Integer Linear Programming) 소프트웨어인 lp_solve3.0을 사용하였다. 시뮬레이션은 Pentium 1GHz 머신에서 수행하였다. 이 lp_solve package 옵션인 CPU time을 이용하여 시뮬레이션 속도를 측정하였다.

<표 4>는 동시 최적화의 결과를 복구 기법에 따라 전체 파장과 거리(마일리지)의 곱들의 합을 나타낸 것이다. Loop LR 기법은 동시 최적화 결과에서 비록 전체 마일리지가 LR보다 10% 증가하였지만 실행시간

표 3. 14-노드 NSFNET 망의 연결요청 행렬 T3

Table 3. Traffic demand matrix T3 for 14-node NSFNET.

	1	2	3	4	5	6	7	8	9	10	11	12	13	14
1														
2	1		1		1	1		2			2	4	1	
3														
4			1									1		
5		7	2				1	6		2	2	8	1	1
6		1		1										
7		2	10		2	1		11		2	2	7	3	
8		6	21	1	3		10		1	4	3	5	1	1
9											1			
10	1	2	4	1	2	1	3	6	1		4	13		2
11		4	1		2	1		6	1	1		2		1
12		3		1	2		1	4	1	3	2			
13		3	1	1	4	1	1	3						1
14														

표 4. T1에 대한 8-노드 망에서의 전체 마일리지와 CPU time

Table 4. Total mileage and CPU time of T1.

	Working mileage	Restoration mileage	Total mileage	Cpu Time (1Ghz)
PR	18330	15120	33450	85.84s
Loop PR	19340	17520	36860	0.7s
LR	17790	16300	34090	0.15s
Loop LR	17960	18530	36490	0.01s

표 5. T2에 대한 14-노드 NSFNET에서의 전체 마일리지와 CPU time

Table 5. Total mileage and CPU time of 14-node NSFNET for T2.

	Working mileage	Restoration mileage	Total mileage	Cpu Time (1Ghz)
PR	2636	1452	4088	>1000s
Loop PR	2590	2665	5255	193.42s
LR	2619	1733	4352	612.87s
loop LR	2462	2313	4775	0.51s

(CPU Time)은 약 7배 이상 빨라졌으며 Loop PR기법은 PR기법 보다 전체 마일리지가 7% 증가하였지만 실행시간(Cpu Time)은 현저하게 빨라졌다는 것을 알 수가 있다.

<표 5>는 연결 요청 행렬 T2에 대한 동시 최적화의 결과를 복구 기법에 따라 전체 파장과 가중치의 곱으로 나타낸 것이다. Loop LR 기법은 최적화 결과에서 비록 전체 마일리지가 LR보다 10% 증가하였지만 실행 시간(CPU Time)은 상당히 감소되었다. Loop PR기법도 PR기법 보다 전체 마일리지가 28% 증가하였지만 실행시간(CPU Time)은 현저하게 빨라졌다는 것을 알 수가 있다.

<표 6>은 <표 5>와 마찬가지로 연결 요청 행렬 T3에 대한 최적화의 결과를 복구 기법에 따라 전체 파장과 가중치의 곱으로 나타낸 것이다. Loop LR 기법은 최적화 결과에서 비록 전체 마일리지가 LR보다 25% 증가하였지만 실행시간(CPU Time)은 약 150배 이상 빨라졌으며 Loop PR기법은 PR기법 보다 전체 마일리지가 50% 증가하였지만 실행시간(CPU Time)은 현저하게 빨라졌다는 것을 알 수가 있다.

<표 4>와 <표 5>, <표 6>에서의 각 기법에 따라 최적값을 얻을 수 있는 실행시간(CPU Time)을 비교하여 보면 <표 4>에서는 Loop PR, Loop LR 기법은 CPU Time은 1초내의 값으로 결과가 나오게 되었다. 그리고 <표 5>와 <표 6>에서는 Loop LR 기법은 PR 기법과 LR 기법보다 빠른 CPU Time을 볼 수 있다. 그러나 Loop PR기법은 트래픽의 분포와 양에 따라서 CPU Time이 LR 기법보다 증가하거나 또는 적어지게 되는데 이 이유는 페루프를 우회하는 복구 경로가 많을수록 CPU Time은 증가하게 되고 페루프내에서 복구되는 경로가 많아지면 당연히 CPU Time은 기존 기법보다 빠르게 된다.

표 6. T3에 대한 14-노드 NSFNET에서의 전체 마일리지와 CPU time

Table 6. Total mileage and CPU time for 14-node NSFNET for T3.

	Working mileage	Restoration mileage	Total mileage	Cpu Time (1Ghz)
PR	3150	2464	5614	>1000s
Loop PR	3479	3881	8426	531.06s
LR	3348	2737	6085	151.07s
loop LR	3212	4437	7649	1.54s

V. 결 론

본 논문에서는 복구 경로를 확보하는 두 개의 새로운 알고리즘을 구현하고 정수 선형 계획법을 제시하였다. 네트워크의 노드가 증가할수록 시간이 O^N 만큼 증가하게 되므로 전체 마일리지와 시간과의 관계를 고려하여 8-노드 망과 14-노드 망에서의 트래픽 연결 요청 행렬에 따라 시간과 전체 마일리지를 비교하였다. 본 논문에서 제시한 Loop LR 알고리즘은 페루프내에서 복구 경로를 확보하는 기법이고 Loop PR 알고리즘은 페루프를 우회하여 복구 경로를 확보고 우회가 여의치 않을 경우에는 페루프내에서 복구 경로를 확보하는 혼합기법이다. 이러한 두 개의 알고리즘을 이용하게 되면 기존의 각 링크마다 복구 경로를 설정해주는 기존 기법은 여러 가지의 장애 발생 경우의 수를 고려하여야 하지만 이 알고리즘은 페루프의 장애만을 고려하여 장애 발생 경우의 수를 최소화 시키게 되므로 CPU time 이 작아지는 결과를 얻게 되었다. 그러나 본 논문에서 제안한 알고리즘은 메쉬 기반의 생존성 WDM망을 구현하기 위해서 필요한 기존의 복구 기법인 LR 기법과 PR 기법보다 최적의 해를 구하기 위한 실행시간(CPU time)이 대폭적으로 감소하였으나 전체 마일리지가 다소 증가하였다는 것을 확인하였다. 이는 CPU time을 줄이기 위해서는 전체 마일리지가 증가하게 되고 전체 마일리지를 최적화시키기 위해서는 CPU time이 증가하게 되는 Trade-off 관계가 있다는 것을 확인하였다.

참 고 문 헌

- [1] B. Mukherjee, "Optical communication networks," McGraw-Hill, NewYork, 1997.
- [2] R. Ramaswami and K. N. Sivarajan, "Design of logical topologies for wavelength-routed optical networks," IEEE J. Select. Areas in Commun., Vol. 14, no. 5, pp. 840~851, June 1996.
- [3] K. Muramkami and H. Kim, "Optimal capacity and flow assignment for self-healing ATM networks based on line and end-to-end restoration," IEEE Trans. Networking, Vol. 6, no. 2, pp. 207~221, Apr. 1998.
- [4] B. Caenegem, W. Parys, F. Truck, and P. Demeester, "Dimensioning of survivable WDM networks," IEEE J. Select. Areas in Commun. Vol. 16, no. 7, Sep. 1998.
- [5] R. Iraschko, M. MacGregor, and W. Grover, "Optimal capacity placement for path restoration in STM or ATM mesh-survivable network," IEEE Trans. Networking, Vol. 6, no. 3, pp. 325~336, Jun. 1998.
- [6] T. H. Wu, "Emerging technologies for fiber network survivability," in IEEE Communications Magazine, vol. 33, no. 2, pp. 58~74, Feb. 1995.
- [7] M. Herzberg, "A decomposition approach to assign spare channels in self-healing network," IEEE Globecom '93, Texas, USA, pp. 1601~1605.
- [8] W. D. Grover, T. D. Blodreau, and B. D. Venable, "Near optimal spare capacity planning in a mesh restorable networks," IEEE Globecom '91, pp. 2007~2012.
- [9] M. Herzberg and A. Utano, "The hop-limit approach for spare-capacity assignment in survivable networks," IEEE/ACM Trans. Networking, Vol. 3, no. 6, pp. 775~784, Dec. 1995.
- [10] Y. Miyao and H. Saito, "Optimal design and evaluation of survivable WDM transport networks," IEEE J. Select. Areas in Commun. Vol. 16, no. 7, Sep. 1998.
- [11] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh network, part I - protection," IEEE Infocom '99, pp. 744~751.
- [12] S. Ramamurthy and B. Mukherjee, "Survivable WDM mesh network, part II - restoration," IEEE Infocom '99, pp. 2023~2030.

저 자 소 개



鄭 榮 哲(正會員)

1981년 : 서울대학교 전자공학과(공학사). 1983년 : 한국과학기술원 전기 및 전자공학과(공학석사). 1992년 : Electrical and Computer Engineering(박사), University of California, Santa Barbara(UCSB).

1981년~1986년 : LG전선기술연구소. 1987년~1988년 : 한국과학기술연구원(KIST). 1992년~1993년 : Opto-electronics Technology Center(UCSB) 연구원. 1993년~현재 : 광운대학교 전자통신공학과 교수. <주관심분야 : 광통신용 소자 및 광통신 시스템>



玄 基 毫(正會員)

1999년 : 경기대학교 전자공학과 학사. 2002년 : 광운대학교 전자통신학과 석사. 2002년 3월~현재 : 한국레이콤 기술연구소에서 “무선 광통신”연구중