

광 다단계 상호연결망의 단일 누화고장에 대한 해석적 고장진단 기법

Analytical Diagnosis of Single Crosstalk-Fault in Optical Multistage Interconnection Networks

김 영 재, 조 광 현
(Young-Jae Kim and Kwang-Hyun Cho)

Abstract : Optical Multistage Interconnection Networks(OMINs) comprising photonic switches have been studied extensively as important interconnecting building blocks for communication networks and parallel computing systems. A basic element of photonic switching networks is a 2×2 directional coupler with two inputs and two outputs. This paper is concerned with the diagnosis of crosstalk-faults in OMINs. As the size of today's network becomes very large, the conventional diagnosis methods based on tests and simulation have become inefficient, or even more, impractical. In this paper, we propose a simple and easily implementable algorithm for detection and isolation of the single crosstalk-fault in OMINs. Specifically, we develop an algorithm for the isolation of the source fault in switching elements whenever the single crosstalk-fault is detected in OMINs. The proposed algorithm is illustrated by an example of 16×16 banyan network.

Keywords : optical multistage interconnection network, crosstalk-fault, fault diagnosis, optical switch

I. 서론

병렬컴퓨터시스템과 통신망 등에서 성능향상을 위해 시스템간의 상호연결을 위한 구조로 다단계 상호연결망(MIN: Multistage Interconnection Network)이 사용된다. 더욱이 현재의 통신시스템에서 요구되는 다양한 서비스 및 처리속도를 충족시키기 위해서 광소자를 사용한 Tbps급의 높은 처리속도와 대역폭을 갖는 광 다단계 상호연결망(OMIN: Optical Multistage Interconnection Network)의 연구가 활발히 이루어지고 있다. 광 통신망에서의 OMIN은 망에 존재하는 교환기 내부의 스위치 연결방법을 제공하고 있다. 따라서 OMIN의 확장은 교환시스템의 대형화를 초래하고 이는 전체 망의 확장으로 이어진다. 그러므로 망의 성능향상과 신뢰도 향상을 위해서는 OMIN시스템의 연구가 필수적이다. 대형화되고 복잡해진 통신망에서는 내부의 망 연결관계를 제공하는 OMIN 또한 그 복잡도가 증가하게 된다. OMIN의 복잡도 증가는 고장 발생시 고장의 유형을 파악하고 위치를 찾아내는 데 어려움을 초래하였다. 이는 통신망의 내고장성 저하로 이어져 전체 망의 성능에도 심각한 문제를 야기시킨다. 그러므로 고장진단기술의 개발은 시스템 성능의 신뢰성을 확보하고 내고장성을 향상시키기 위해서 반드시 선행되어야 할 과제이다. OMIN은 2×2 광 스위치(optical switch)들의 상호연결로 구성되어 있다. $N \times N$ OMIN은 N 개의 입력과 N 개 출력 사이의 연결을 제공하며 $m(= \log_2 N)$ 개의 단

(stage)으로 이루어져 있다. 각 단은 $N/2$ 개의 2×2 광 스위치들을 포함한다[1][2].

광 스위치를 사용하는 OMIN에서 가장 큰 문제점은 누화(crosstalk)에 의해 발생하는 고장이다. 누화고장(crosstalk-fault)은 인접의 빛의 간섭현상이 원인이다. 이것은 광소자를 사용하는 모든 매체에서 빈번히 발생하며, 광 스위치 소자의 채널을 통해 전송되던 신호가 이웃한 다른 채널을 통해서도 나타나는 현상이다. 그림 1은 광 스위치에서 발생한 누화고장의 예를 나타낸다[3][4].

OMIN의 신뢰도를 높이기 위해서는 누화고장에 대한 대책을 마련해야 한다. 시스템에서 고장이 일어나는지 계속 감시하면서 고장이 발생했을 때는 이를 정확하게 진단하여 그때에 필요한 조치를 취해 주어야 한다. 고장을 조기에 진

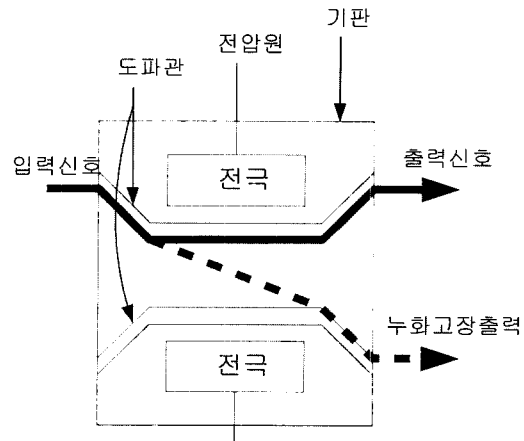


그림 1. 누화고장이 발생한 광 스위치.
Fig. 1. Crosstalk-fault in an optical switching element.

접수일자 : 2001. 8. 6., 수정완료 : 2001. 12. 28.
김영재, 조광현 : 울산대학교 전기전자정보시스템공학부(kyj@sys.ulsan.ac.kr/ckh@mail.ulsan.ac.kr)
※ 이 논문은 2001년도 한국학술진흥재단의 지원에 의하여 연구가 되었음(KRF-2001-041-E00279).

단하는 기술은 OMIN시스템의 내고장성 향상을 통해 시스템의 신뢰도를 높이기 위해서 반드시 필요하다. 이에 본 논문은 OMIN을 구성하고 있는 광 스위치 소자에서 발생하는 누화고장을 진단하기 위한 알고리즘을 중점적으로 다룬다. 지금까지 누화고장 진단을 위해서는 주로 테스트기법을 사용하여 왔다[5][6]. 테스트기법은 수 차례의 테스트를 통해 고장을 진단하는 기법이며, 현재는 이러한 테스트기법으로 OMIN에서 발생하는 다중고장을 진단해내고 있다. 한편, 통신망시스템의 확장을 위해서는 OMIN시스템의 확장을 필요로 하는데 이는 또한 시스템의 복잡도를 증가시킨다. 시스템 복잡도의 증가는 고장진단을 위한 테스트 횟수의 증가로 이어져 수 차례의 테스트 이후에도 고장을 찾지 못하는 경우가 발생할 수 있다. 그러므로 테스트기법에 대한 연구는 주로 고장발생시 고장진단을 위한 테스트의 횟수를 줄이는 방향으로 진행되어 지고 있다. 그러나 테스트기법의 한계점을 극복하고 효과적인 고장진단을 위해서는 새로운 해석적 접근방법에 의한 고장진단기법의 연구가 필요하다. 이에 본 논문에서는 OMIN의 단일 누화고장 발생시 시스템의 입력과 출력 정보만을 가지고 단 한번의 알고리즘 수행으로도 효과적으로 고장을 진단할 수 있는 방법론을 제안한다.

II. 광 다단계 상호연결망과 누화고장 모델

1. 광 다단계 상호연결망

OMIN을 구성하는 2x2 광 스위치는 제어비트(스위치의 라우팅 정보)를 참조하여 제어비트가 0이면 상위 포트(port)로, 1일 때는 하위 포트로 각각 라우팅한다. 이때 제어비트는 광 스위치 내부에서 라우팅을 위한 정보로 사용되며, 출력포트별 나타내는 식별자이다. 즉, 제어비트는 OMIN에서 목적지(destination)정보를 가지고 있으며, 각 단에서는 광 스위치의 라우팅을 결정하게 된다. 그림 2는 광 스위치의 4가지 동작형태를 나타낸다[6][7].

OMIN은 광 스위치의 연결관계에 따라 뱀안(banyan), 오메가(omega), 큐브(cube)망 등으로 망의 종류를 구분하고 있다[7][8]. 본 논문에서는 이 가운데 가장 널리 쓰이는 뱀안스위치 구성을 갖는 망을 대상으로 한다. 그림 3에 8개의 입력포트와 8개의 출력포트를 갖는 8x8 뱀안 망의 예가 나타나 있다. 8x8 OMIN은 3(= log₂8)단으로 구성되어 있으며 각 단에는 총 4(8/2)개의 스위치를 포함하고 있다. 뱀안 망에서 단의 상호연결은 1-비트 순환 우 천이(1-bit circular right shift)방식을 사용한다. 이 방식은 입력원(source) 주소의 0번째와 k번째 비트를 바꾸어 목적지 주소를 얻는다. 그러므로 스위치를 연결하기 위한 링크함수 L은 $L(b_{m-1} \dots b_{k+1} b_k$

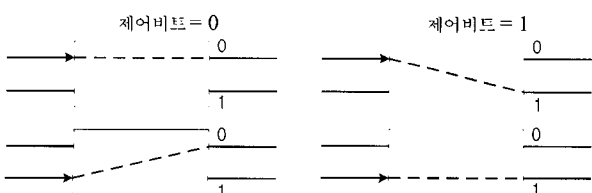


그림 2. 광 스위치의 동작형태.
Fig. 2. Operation of an optical switch.

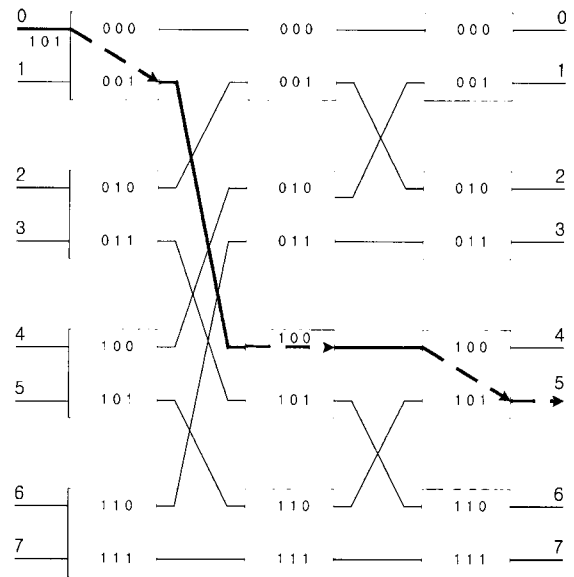


그림 3. 8x8 뱀안 망.
Fig. 3. 8x8 banyan network.

$$b_{k-1} \dots b_1 b_0 = b_{m-1} \dots b_{k+1} b_k b_{k-1} \dots b_1 b_0 \text{ 된다}[7].$$

그림 3에서 제어비트 101(십진수 5)을 가진 신호가 0번 포트에 입력되고 있다. 이 신호는 출력포트 #5에 전송되어야 할 신호이다. 제어비트는 가장 왼쪽비트가 최상위 비트이며 왼쪽에서 오른쪽으로 비트의 순서가 정해진다. 첫 번째 단의 2x2 광 스위치에서는 제어비트 101의 최상위 비트 1에 의해 하위 포트, 두 번째 단에서는 제어비트의 2번째 비트 0에 의해 상위포트로, 세 번째 단은 제어비트의 3번째 비트 1에 의해 하위포트로 라우팅하여 목적지까지 신호가 전송된다.

III. 2 누화고장이 발생한 OMIN

그림 4는 그림 1과 같은 누화고장이 발생한 광 스위치를 포함하고 있는 OMIN의 예를 보여준다. 즉, 입력포트 #4에

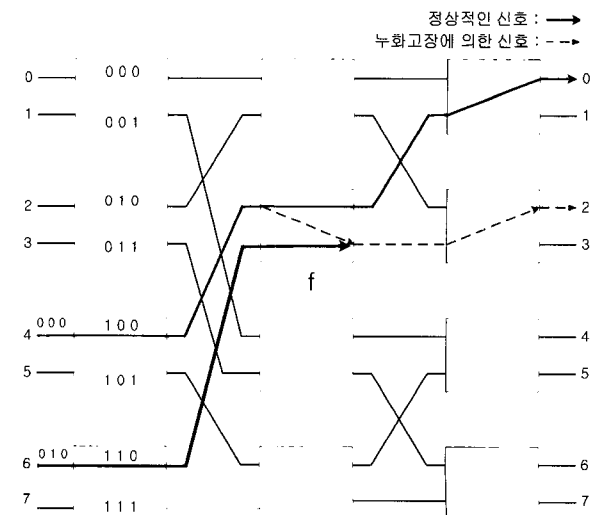


그림 4. OMINs에서의 누화고장 경로.
Fig. 4. Illustration of the crosstalk-fault path in OMINs.

서 출력포트 #0으로 신호를 전송하고 있으나 스위치 f에서 누화고장이 발생하여 출력포트 #2로 원치 않는 출력이 발생하였다. 그리고 입력포트 #6에서 출력포트 #2로 신호를 보내는 도중에 스위치 f에서 발생한 누화 신호로 인하여 정상적인 신호의 흐름을 방해하거나 막고 있다. 이로 인하여 데이터흐름을 방해하는 충돌(conflict) 문제와 한정된 망 자원의 손실 등이 유발된다. 이러한 문제를 해결하기 위해서는 반드시 정확하고 신속한 고장진단기법이 요구된다.

IV. 3 누화고장 진단을 위한 광 스위치의 대수적 모델링

제어비트 c_k (k 번째 단의 제어비트)를 갖는 광 스위치에 누화고장이 발생했을 때, 광 스위치를 그림 5와 같이 모델링하여 보자.

그림 5에서 입력측의 상위포트를 하위포트를 각각 α, β 라 하고, 출력측의 상위포트를 하위 포트는 γ, δ 라고 하여, OMIN의 k 단에 위치한 광 스위치의 입력행렬(input matrix)은 $M_{i,k} = \begin{bmatrix} \text{상위포트} \\ \text{하위포트} \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}$ 로 표기한다. 만약 스위치에 입력이 단일 신호 일 때 α, β 는 0 또는 1의 값을 가지며 입력 신호가 존재하면 1, 그렇지 않으면 0이다. 즉, 항상 $\alpha + \beta = 1$ 이 된다. 그림 4와 같은 OMIN에서 신호의 입력이 시작되는 입력포트의 주소를 통해 이 신호가 전송될 경로에 위치한 각각의 스위치에 대한 입력행렬 $M_{i,k}$ 를 알 수 있다. 이때 OMIN의 임의의 시작지 포트 ϕ 를 통해 입력된 신호의 전송 경로에 위치한 스위치들의 전체 입력행렬(total input matrix) Φ_T 는 다음과 같이 정의된다.

정의 1 : $\Phi_T = [M_{i,1} \ M_{i,2} \ \dots \ M_{i,k} \ \dots \ M_{i,m}]$ 이며, $2 \times m$ 행렬이다. 이때 시작지 포트 ϕ 의 2진 코드워드를 $\phi = \phi_{m-1} \phi_{m-2} \dots \phi_1 \phi_0$ 라 하면 Φ_T 는 하기와 같은 논리적 연산을 통해 결정된다:

$$\begin{aligned} \phi &= \phi_{m-1} \phi_{m-2} \dots \phi_1 \phi_0 \\ \Rightarrow \bar{\phi} &= \psi'_{m-1} \psi'_{m-2} \dots \psi'_1 \psi'_0 \end{aligned} \quad (1)$$

$$\begin{aligned} inv(\bar{\phi}) &= \psi'_0 \psi'_1 \dots \psi'_{m-2} \psi'_{m-1} \\ \Rightarrow inv(\bar{\phi}) &= \psi''_0 \psi''_1 \dots \psi''_{m-2} \psi''_{m-1} \end{aligned} \quad (2)$$

$$\begin{aligned} \Phi_T &= \begin{bmatrix} inv(\bar{\phi}) \\ inv(\psi') \end{bmatrix} \\ &= \begin{bmatrix} \psi'_0 & \psi'_1 & \dots & \psi'_{m-2} & \psi'_{m-1} \\ \psi''_0 & \psi''_1 & \dots & \psi''_{m-2} & \psi''_{m-1} \end{bmatrix}. \end{aligned} \quad (3)$$

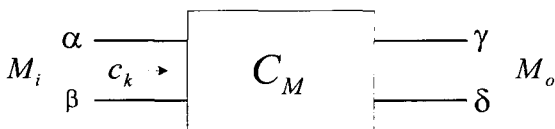


그림 5. 광스위치의 모델링.
Fig. 5. Modeling of an optical switch.

여기서 $\bar{\phi}, inv(\bar{\phi})$ 는 각각 ϕ 에 논리연산 NOT을 취한것과 $\bar{\phi}$ 의 2진 코드워드를 역으로 재배열한 것이다.

광 스위치의 출력행렬(output matrix)은 $M_o = [\text{상위포트} \text{ 하위포트}] = [\gamma \ \delta]$ 로 한다. 여기서 γ, δ 값은 정상적인 출력이면 1, 비정상적인 출력, 즉 누화고장이 발생한 출력이면 0으로 표기한다. 그리고 제어비트 c_k 와 γ, δ 의 관계는 $\gamma = 1 - c_k, \delta = c_k$ 라 한다.

정의 2 : 누화고장 행렬(crosstalk-fault matrix) C_M 은 다음과 같이 정의된다:

$$\begin{aligned} C_M &= M_{i,k} \cdot M_o = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \cdot [\gamma \ \delta] = \begin{bmatrix} \alpha\gamma & \alpha\delta \\ \beta\gamma & \beta\delta \end{bmatrix} \\ C_M &= \begin{bmatrix} \alpha(1-c_k) & \alpha c_k \\ \beta(1-c_k) & \beta c_k \end{bmatrix}. \end{aligned} \quad (4)$$

이다.

정리 1 : 입력행렬 M_i 와 제어비트 c_k 를 가지는 광 스위치에 누화고장이 발생하면 출력행렬 $M_o = M_{i,k}^T \cdot C_M$ 이 된다.

증명 : 정의 2를 이용하여,

$$\begin{aligned} M_o &= M_{i,k}^T \cdot C_M = [\alpha \ \beta] \cdot \begin{bmatrix} \alpha(1-c_k) & \alpha c_k \\ \beta(1-c_k) & \beta c_k \end{bmatrix} \\ &= [\alpha^2(1-c_k) + \beta^2(1-c_k) \quad \alpha^2 c_k + \beta^2 c_k] \\ &= (\alpha^2 + \beta^2) [1 - c_k \quad c_k] \end{aligned} \quad (5)$$

이다. 위의 결과에서 $\alpha^2 + \beta^2 = 1$ 이므로, 식을 다시 정리하면 $[1 - c_k \quad c_k] = [\gamma \ \delta]$ 이다. 이것은 그림 5의 출력행렬 M_o 와 일치함을 알 수 있다. ■

그림 1과 같은 광 스위치에서 누화고장이 발생하는 경우 그림 6과 같은 4가지 유형으로 분류된다[7].

고장이 발생한 광 스위치는 $M_{i,k}$ 와 c_k 를 통해 고장유형을 식별해 낼 수 있다. 즉, 누화고장 발생시 그림 6의 고장유형 번호(fault-type number)는 $M_{i,k}^T = [\alpha \ \beta]$ 일 때 제어비트가 c_k 이면 고장유형 번호의 이진수 표현 $N_B = \beta c_k$ 이다. 이를테면, $M_{i,k}^T = [0 \ 1], c_k = 0$ 이면 $N_B = 10$ 이며, 십진수 표현인 $N_D = 2$ 가 되어 두 번째 고장유형에 해당된다. OMIN에서 임의의 광 스위치에 고장이 발생했을 때 그림 6에서 보

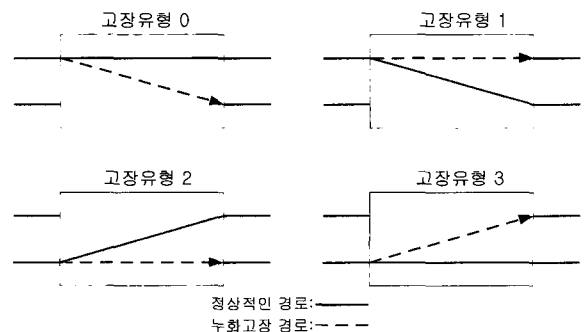


그림 6. 4가지 유형의 누화고장.
Fig. 6. 4 types of crosstalk-faults.

여진 4가지 누화고장 유형은 정의 1, 정의 2 그리고 정리 1에 의해서 알아낼 수 있다. 그리고 광 스위치의 $M_{i,k}$ 와 c_k 를 통해 C_M, M_o 및 N_B 를 알아낼 수 있다. 만일 $c_k=0$ 이며 상위 포트에 입력이 가해진 입력행렬 $M_{i,k}^T = [\alpha \ \beta] = [1 \ 0]$ 인 임의의 광 스위치에 누화고장이 발생했다면, 이때 광 스위치의 C_M, M_o, N_B 는 각각 다음과 같다:

- 정의 2에 의해서 $C_M = \begin{bmatrix} \alpha(1-c_k) & \alpha c_k \\ \beta(1-c_k) & \beta c_k \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$ 이 된다.

된다.

- 정리 1에 의해서

$$M_o = M_{i,k}^T \cdot C_M = [1 \ 0] \cdot \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} = [1 \ 0]$$

이를 통해 상위포트는 정상적인 출력이나 하위포트는 누화고장에 의한 출력임을 알 수 있다.

- 고장유형 번호 $N_B = \beta c_k = 00 \Leftrightarrow N_D = 0$ 임을 알 수 있다.

- 이 광 스위치는 출력의 하위포트에 누화고장에 의한 신호의 출력이 발생되었으며, 고장유형 0인 고장임을 알 수 있다.

이 결과는 알고리즘 상에서 고장유형을 찾아내고 진단하기 위한 유용한 정보로 사용된다. 그리고 광 스위치에 고장이 발생하면, 이와 같은 과정을 통해 고장의 종류에 따른 고장해석이 가능하다.

V. 가상 라우팅 테이블과 OMIN의 시스템 행렬

1. 가상 라우팅 테이블

OMIN에서 유기적으로 연결되어 있는 광 스위치는 제어 비트에 의해서 독립적으로 동작하여 시스템의 확장을 용이하게 한다. 일반적으로 2×2 광 스위치는 그림 2에서 보는 바와 같이 4가지의 동작형태를 갖는다. 이러한 동작형태는 다시 그림 7에서와 같이 2가지의 동작형태로 축약될 수 있으며 다음과 같이 정의된다[7][8].

정의 3 : 제어비트에 의해서 스위치가 병렬연결(parallel connection: P)일 때 스위치 상태값은 0을 교차연결(cross connection: X)이면 스위치의 상태값은 1을 갖는다:

$$P \begin{cases} \alpha(c_k) = \overline{c_k} \\ \beta(c_k) = c_k \end{cases}, \quad X \begin{cases} \alpha(c_k) = c_k \\ \beta(c_k) = \overline{c_k} \end{cases} \quad (6)$$

이 2가지 스위치 연결형태만을 고려하여 OMIN에서 새로운 가상 라우팅 경로(VRP: Virtual Routing Path)를 설정할 수 있으며, 이를 이용하여 오프라인(off-line)상에 새로운 가상 라우팅 테이블(VRT: Virtual Routing Table)을 구성할 수 있다. 이 VRT의 간소화를 위하여 스위치의 두 가지 연결상

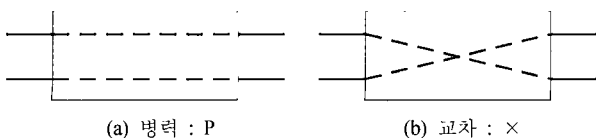


그림 7. 스위치의 연결상태.

Fig. 7. Connection status of a switch.

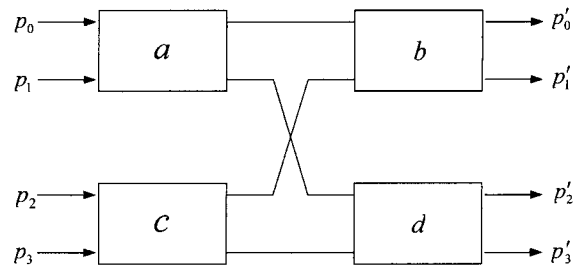


그림 8. 스위치에 라벨 부여하기.

Fig. 8. Labeling of an optical switch.

태만을 고려하였다. 그리고 고장진단시에는 스위치의 연결 형태에 관한 정보와 위치정보를 제공하는 역할을 한다. 다음은 그림 8의 4×4 OMIN의 예를 통해 VRT를 생성하는 과정을 나타낸 것이다.

단계 1 : 그림 8과 같이 OMIN에 존재하는 모든 스위치에 라벨(label)을 부여한다. 이 라벨집합은 $C_o = \{a, b, c, d, \dots\}$ 으로 나타내며 이때 라벨원소의 전체개수는 $2^{m-1} \times m$ 이다.

단계 2 : 입력원에서 목적지까지 사용되어지는 광 스위치의 조합(permutation)을 라벨을 이용하여 표기한다. 이를테면 그림 8의 경우 다음과 같다:

$$C_o = \begin{pmatrix} a_{00} & a_{01} & a_{02} & a_{03} \\ a_{10} & a_{11} & a_{12} & a_{13} \\ a_{20} & a_{21} & a_{22} & a_{23} \\ a_{30} & a_{31} & a_{32} & a_{33} \end{pmatrix} = \begin{pmatrix} (a, b) & (a, b) & (c, b) & (c, b) \\ (a, b) & (a, b) & (c, b) & (c, b) \\ (a, d) & (a, d) & (c, d) & (c, d) \\ (a, d) & (a, d) & (c, d) & (c, d) \end{pmatrix}$$

여기서, a_{ij} 는 입력원 j 에서 목적지 i 로의 경로를 표현하며, C_o 는 이러한 경로에 사용된 스위치의 라벨을 의미한다.

단계 3 : 정의 3을 이용하여 입력원에서 목적지까지의 스위치 상태값(C_{ov})을 다음과 같이 구한다. 입력원과 목적지의 2진 코드를 각각, $j = s_{m-1}s_{m-2}s_{m-3} \dots s_1s_0$, $i = d_{m-1}d_{m-2}d_{m-3} \dots d_1d_0$ 이라 할 때 $C_{ov} = \overline{(j \oplus i)}$ 이 된다. 이때 단계 2의 C_o 에 대한 모든 C_{ov} 는, 그림 8의 경우

$$C_{ov} = \begin{pmatrix} (0, 0) & (1, 0) & (0, 1) & (1, 1) \\ (0, 1) & (1, 1) & (0, 0) & (1, 0) \\ (1, 0) & (0, 0) & (1, 1) & (0, 1) \\ (1, 1) & (0, 1) & (1, 0) & (0, 0) \end{pmatrix}$$

이다. 예를 들면, 입력원 $j=2$ 를 $i=3$ 으로 연결하기 위한 스위치 상태 $C_{ov} = \overline{(10 \oplus 11)} = 10$ 이 된다. 이는 2번($j=2$) 포트에 입력된 신호는 라벨이 c인 스위치에서 X상태로 라우

표 1. 4×4 OMIN 가상 라우팅 테이블.
Table 1. Virtual routing table of 4×4 OMIN.

a_{ij}	C_o	C_{ov}	a_{ij}	C_o	C_{ov}
a_{00}	(a, b)	0 0	a_{10}	(a, b)	0 1
a_{01}	(a, b)	1 0	a_{11}	(a, b)	1 1
a_{02}	(c, b)	0 1	a_{12}	(c, b)	0 0
a_{03}	(c, b)	1 1	a_{13}	(c, b)	1 0

a_{ij}	C_o	C_{ov}	a_{ij}	C_o	C_{ov}
a_{20}	(a, d)	1 0	a_{30}	(a, d)	1 1
a_{21}	(a, d)	0 0	a_{31}	(a, d)	0 1
a_{22}	(c, d)	1 1	a_{32}	(c, d)	1 0
a_{23}	(c, d)	0 1	a_{33}	(c, d)	0 0

팅 된 후, P상태인 스위치 d를 통해 목적지의 3번($i=3$) 포트까지 신호가 전송된다. C_{ov} 는 목적지 정보에 상응하는 새로운 라우팅정보인 VRT가 되며, 고장진단시 알고리즘에 광 스위치의 상태정보 및 라우팅 경로를 제공한다.

단계 4: 이상의 과정을 통해 4×4 OMIN의 VRT를 표 1과 같이 완성할 수 있다.

위와 같은 과정을 통하여 확장된 $n \times n$ OMIN에서의 VRT생성도 가능하다.

VI. 2 OMIN의 시스템 행렬을 통한 데이터의 단일 누화고장 모니터링

어떤 시스템을 해석하기 위해서 일반적으로 사용되는 정보는 입·출력에 관한 내용이다. OMIN에서는 데이터가 전송되길 원하는 목적지는 출력포트의 주소가 되고, 데이터 전송이 시작되는 곳은 입력포트 주소가 된다. 이러한 입·출력 정보를 가지고 OMIN의 특성을 나타내는 행렬을 표현할 수 있으며, 현재 진행된 라우팅의 상황을 모니터링할 수 있다. 단일 신호 입력시 $n \times n$ OMIN의 n 개의 포트에 대한 입력행렬 P 와 출력행렬 P' 이 존재할 때 시스템행렬 S_M 은 다음과 같이 정의된다:

정의 4:

$$S_M = P \cdot P' \tag{7}$$

$$= [p_0 \ \dots \ p_j \ \dots \ p_{n-2} \ p_n]_r \begin{bmatrix} p'_0 \\ \vdots \\ p'_i \\ \vdots \\ p'_{n-2} \\ p'_{n-1} \end{bmatrix}$$

이다.

이때 $p_j \cdot p'_i = p_{ij}$ 임을 알 수 있다. 그리고 p_j, p'_i 은 입력 포트나 출력포트에 신호가 존재하면 1 없으면 0이다.

S_M 의 특징은 다음과 같다:

- 신호가 정상적으로 전송되어 OMIN에 어떠한 고장도 존재하지 않을 때, S_M 에 원소 p_{ij} 가 존재하면 i 행과 j 열에는 p_{ij} 이외의 어떠한 원소도 존재할 수 없다. 즉, 각각의 행과 열에는 하나 이상의 원소는 존재할 수 없다. 이는 다음과 같은 이유에 기인한다.

- 각 열(column)의 두 개 이상의 원소가 1이면 OMIN에 누화고장이 발생한 경우이다. 즉, 서로 다른 신호가 동일한 입력을 통해 전송이 된 경우이다.

- 각 행(low)의 두 개 이상의 원소가 1이면 OMIN에 충돌(conflict)이 발생한 경우이다. 이것은 서로 다른 신호가 동시에 동일한 목적지를 향해 전송이 된 경우이다. 그리고 광 네트워크에서는 종전의 네트워크와 달리 파장이라는 하나의 새로운 차원을 제공하기 때문에 충돌에 대한 해결법은 이를 이용한 다른 접근법이 필요하다. 그러므로 본 논문에서는 이러한 충돌에 의한 고장은 고려하지 않는다.

S_M 은 OMIN의 입·출력이 존재하는 입·출력포트의 번호를 계속 관찰하면서 시스템의 고장유무 및 경로에 관한 정보를 p_{ij} 의 행과 열이 동일한 a_{ij} 를 호출함으로써 VRT에 제공하게 된다.

VII. 고장진단 알고리즘

본 절에서는 앞 절에서 제시한 VRT와 S_M 을 이용하여 단일 누화고장 발생시 OMIN에서 고장의 위치를 찾아내고, 고장의 유형을 진단하는 알고리즘을 제시한다. 제안하는 고장진단과정을 간단히 기술하면, 모든 입·출력 정보를 계속 관측하면서 P 와 P' 을 구성하여 S_M 을 만든 후 모든 p_j, p'_i 와 p_{ij} 의 내용을 포함하고 있는 S_M 을 고장진단 알고리즘에 통보한다. 알고리즘은 고장이 없으면 0을, 고장이 존재하면 1을 시스템에 알려준다. 그리고 고장진단 알고리즘은 a_{ij} 를 이용하여 VRT의 내용을 참조한 후 고장의 유형 및 위치를 찾아낸다. 이러한 과정을 블록 다이어그램으로 표현하면 그림 9와 같다.

그리고 블록도에 포함된 고장진단 알고리즘을 흐름도를 사용하여 상세히 표현하면 그림 10과 같다. $n \times n$ OMIN에서 임의의 입력포트 p 의 십진수 포트번호 값이 ϕ 인 곳으로

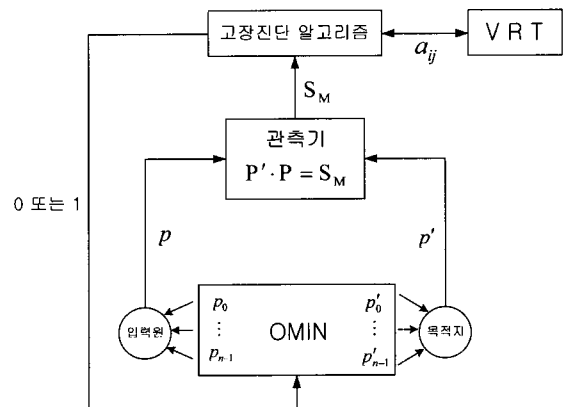


그림 9. 고장진단과정의 블록다이어그램.
Fig. 9. Block diagram of the fault diagnosis procedures.

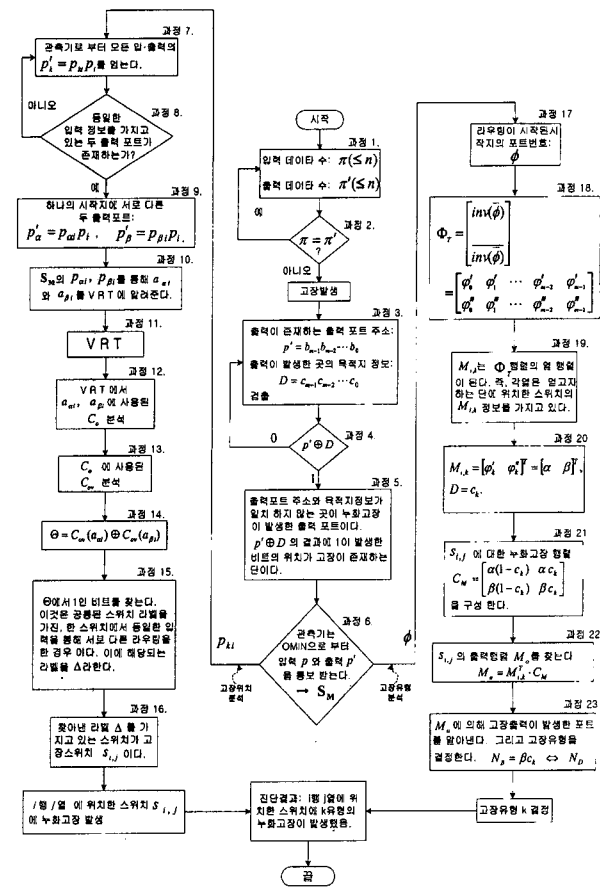


그림 10. 고장진단 알고리즘의 흐름도.
Fig. 10. Flow chart of the fault diagnosis algorithm.

신호가 입력될 때, 이 신호의 목적지 정보를 D 라고 하자. 이때, ϕ 로 입력된 신호가 D 의 정보에 의해 최종적으로 도착한 출력포트의 번호를 p' 라 하면, 정상적인 신호의 전송이 이루어졌을 때 D 와 p' 의 포트번호는 일치할 것이다. 그러나 어느 한 스위치라도 고장이 발생하여 신호가 목적지까지 전송되지 않았다면 D 와 p' 의 포트번호는 서로 다른 값을 가지게 될 것이다.

그림 10의 고장진단 알고리즘은 크게 3부분으로 구성되어 있다. 첫 번째 부분은 과정 1에서 과정 6까지로 OMIN의 고장 발생여부를 파악한다. 두 번째 부분은 과정 7에서 과정 16까지로 고장이 발생한 광 스위치의 위치를 찾는다. 그리고 세 번째 부분은 과정 17에서 과정 23까지로 고장의 유형을 판별하게 된다. 이를 상세히 설명하면 다음과 같다. 먼저 과정 1에서는 그림 9에서 보는 바와 같이 입력되는 모든 데이터의 개수($\pi \leq n$)와 이들 데이터의 입력포트 p 를 파악한다. 그리고 시간 t_d 이후 출력포트에서도 출력된 데이터 ($\pi' \leq n$)의 개수와 출력이 존재하는 포트번호 p' 를 확인한다. 이때, t_d 는 입력된 데이터가 목적지까지 도달하는 시간으로 고려하지 않아도 될만큼 아주 짧은 시간으로 간주한다. 그리고 과정 2에서는 π 와 π' 을 비교하여 일치하면 고장이 존재하지 않는 것이며, 그렇지 않으면 OMIN의 시스템에 고장이 발생한 것이다. 고장이 존재함을 통보받은 과정 3에

서는 OMIN에서 출력이 존재하는 출력포트 주소 p' 과 이때 출력된 데이터의 목적지 정보 D 를 확인한다. 과정 4에서는 $p' \oplus D = (b_{m-1}b_{m-2} \dots b_0) \oplus (c_{m-1}c_{m-2} \dots c_0)$ 를 수행한다. 이를 수행한 연산결과 분석을 통해 1이 발생한 비트의 위치가 OMIN에서 고장이 존재하는 단이 된다. 과정 6은 과정 5를 통해 알게된 p 와 p' 를 그림 9에 보여진 블록도의 관측기에 통보하여 S_M 을 생성한다. 과정 7에서 과정 9는 S_M 으로부터 동일한 입력정보를 가지고 있는 두 출력포트 $p'_a = p_{ai}p_i$ 와 $p'_b = p_{bi}p_i$ 를 찾는다. 이때 얻은 정보를 통해 과정 10에서는 S_M 으로부터 p_{ai} , p_{bi} 를 통해 행과 열이 동일한 두 원소 $a_{a,i}$ 와 $a_{b,i}$ 를 검출하여 VRT에 알려준다. 이때, 3절에서 제시한 방법과 동일한 과정을 통해 현재 OMIN에 맞는 VRT를 오프라인으로 생성해 낸다. 그리고 과정 11에서 13까지는 VRT에서 a_{ai} 와 a_{bi} 에 사용된 C_o 와 C_{oo} 를 모두 분석한다. 분석한 결과를 통해 과정 14에서는 $\Theta = C_{ov}(a_{ai}) \oplus C_{ov}(a_{bi})$ 의 연산을 수행한다. 과정 15에서는 앞에서 수행한 연산의 결과를 분석한다. Θ 에서 결과 값이 1인 비트의 위치를 찾고 이에 해당되는 스위치의 라벨을 확인한다. 이것은 동일한 입력포트를 통해 입력된 데이터가 임의의 광 스위치에서 서로 다르게 라우팅이 된 경우이다. 즉, 공통된 광 스위치 라벨을 가진 한 스위치가 누화고장으로 인해 두 출력포트에 모두 데이터를 전송한 것이다. 이때 해당되는 라벨을 L 라 하면, 이 라벨을 가지고 있는 스위치가 고장스위치이다. 과정 16에서는 이 라벨정보를 통해 고장이 발생한 광 스위치의 위치를 식별하게 된다. 과정 17은 S_M 에서 입력포트 p 를 통해 데이터의 전송이 시작된 포트번호를 ϕ 라 할 때, 과정 18에서 정의 1을 이용하여 전송할 데이터가 OMIN을 통해 전송될 때 사용될 모든 광 스위치의 전체 입력행렬 Φ_T 를 생성한다. 그리고 과정 19와 과정 20에서는 과정 5에서 분석된 고장 단의 위치정보를 통해 Φ_T 에서 고장이 발생한 광 스위치의 입력행렬 $M_{i,k}$ 를 찾는다. 이때, Φ_T 의 열은 OMIN의 단의 정보와 동일하므로 연고자 하는 단에 존재하는 광 스위치의 $M_{i,k}$ 는 Φ_T 에서 k 열에 해당하는 행렬이 된다. 이를 통해 과정 20에서는 해당 고장 스위치의 $M_{i,k}$ 를 알게 되고, 과정 21에서는 정의 2를 이용하여 C_M 을 얻게 된다. 과정 22에서는 정리 1에 의해 M_o 를 알게 되며, 이를 분석하여 고장유형을 파악하게 된다. 이상의 알고리즘은 주로 논리적 연산을 통해 수행되므로, 고장발생시 고장의 위치 및 유형을 정확히 파악하여 신속한 고장진단을 가능하게 한다.

VIII. 누화고장진단의 예

다음은 앞 절에서 제안한 알고리즘을 이용하여 단일 누화고장이 발생한 16x16 배안 망에서 고장진단 과정의 예를 보인다. 입력포트 #4에 목적지 정보 14(1110)를 갖는 신호를 입력하였을 때, 출력포트에서는 #12, #14에서 각각 관측기를 통해 출력이 관찰되었다고 가정하자. 이상의 정보로부터 고장진단 과정을 따라서 고장원인과 유형을 파악해 보자. 첫 번째 부분은 OMIN의 고장발생 여부를 파악하는 단계로

서 과정 1에서 과정 6까지이다:

1) 관측기에서는 입·출력 신호의 수를 비교하여 시스템의 고장발생여부를 분석한다. 입력데이터 수 $\pi=1$, 출력 데이터 수 $\pi=2$ 이므로, $\pi=\pi$ 를 만족시키지 못하므로 이 시스템은 고장이 발생했음을 알 수 있다(알고리즘의 과정 1에서 과정 2까지이다).

2) 출력이 존재하는 출력포트 번호와 목적지 정보를 비교하여 누화고장에 의한 출력과 OMIN에서 고장이 발생한 단을 찾는다(과정 3에서 과정 5까지이다). $p'_{12}=1100$, $p'_{14}=1110$, $D=1110$ 이므로, $p'_{12} \oplus D_{14} = 0010$, $p'_{14} \oplus D_{14} = 0000$ 이다. p'_{12} 출력이 누화고장에 의한 출력이며, 세 번째 단에서 고장이 발생했음을 알 수 있다.

3) 관측기는 모든 입·출력을 수집하여 S_M 을 구성한다(과정 6이다). $p'_{12} \cdot p_4 = p_{12,4}$, $p'_{14} \cdot p_4 = p_{14,4}$ 에 의해 두 개의 원소를 갖는다.

다음 두 번째 부분에서는 고장의 위치를 VRT를 통해 찾는다:

4) VRT는 S_M 의 $p_{12,4}$, $p_{14,4}$ 를 통해 $a_{12,4}$, $a_{14,4}$ 에 쓰인 C_o , C_{ov} 를 분석한다(알고리즘의 과정 7에서 16까지이다). $(i, v, A, G) \oplus (i, v, A, B) = 1100 \oplus 1110 = 0010$ 이므로 세 번째 라벨 A가 다르다. 이를 통해 라벨 A를 가진 7행 3열의 스위치에 고장이 발생하였음을 알 수 있다.

마지막 세 번째 부분은 고장 유형을 진단하는 부분이다:

5) 정의 1에 의해 신호가 입력된 입력포트를 통해 사용될 전체 스위치의 전체 입력행렬 Φ_T 를 만든다(과정 17에서 과정 23까지이다). $\phi=4$ 이고, 4의 이진수는 0100이 된다. $\bar{\phi} = 1011$ 이고, 이 1011을 다시 역 배열하면 $inv(\bar{\phi}) = 1101$ 이 된다. 따라서 $inv(\bar{\phi}) = \overline{1101} = 0010$ 이 된다. 이때 Φ_T 는

$$\Phi_T = \begin{bmatrix} inv(\bar{\phi}) \\ inv(\bar{\phi}) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \text{이다.}$$

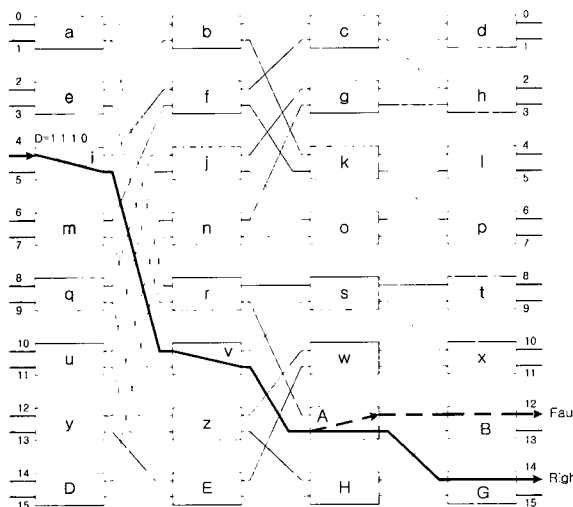


그림 11. 예에서 사용된 16×16 뱀안 망의 고장진단 결과.
Fig. 11. Fault diagnosis result of the example 16×16 banyan network.

6) 위 2)를 통해 세 번째 단에서 고장이 발생했음을 알았으므로 세 번째 단에 위치한 고장스위치의 입력행렬 $M_{i,3} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ 이 된다.

7) $S_{7,3}$ 에 대한 $M_{i,3} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$, $D=1110$ 에서 $c_3 = 1$ 이므로, $C_M = \begin{bmatrix} \alpha(1-c_k) & \alpha c_k \\ \beta(1-c_k) & \beta c_k \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$ 이다.

8) $M_o = M_{i,3}^T \cdot C_M = [0 \ 1] \cdot \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix} = [0 \ 1]$ 이다.

이상의 과정을 통해 고장이 발생한 스위치는 7행 3열이며, 스위치의 상위포트에서 누화고장이 발생했음을 알 수 있다. 그림 11은 알고리즘에 의한 고장진단 결과를 도식화한 것이다.

그림 12는 이러한 일련의 과정을 모의실험을 통해 확인한 결과를 나타낸 것이다. 모의실험에서는 MATLAB(Ver. 6.0)을 이용하여 그림 10에서 보여진 과정 3에서 과정 23까지를 구현하였다. 그림 12(d)는 위에서 예시한 고장의 모의실험 결과이며, (a)-(c)는 단일 입력신호로부터 두 개의 출력신호가 검출된 경우를 임의로 발생시켜 각 유형별로 모의실험한 결과이다. 그림 12의 모의실험에서 16×16 OMIN의 입·출력 포트번호는 가장 상위포트번호를 1, 가장 하위포트번호를 16으로 하였다. 모의실험은 OMIN에서 신호가 입·출력된 포트의 정보와 이에 대한 고장스위치의 위치 및 고장 유형을 보여주고 있다. 모의실험 결과화면에 나타난 5번째 줄의 내용은 16×16 OMIN에 사용된 2×2스위치 8개와 4개의 단을 보여주고 있다. 이때 열은 단의 정보이고 행

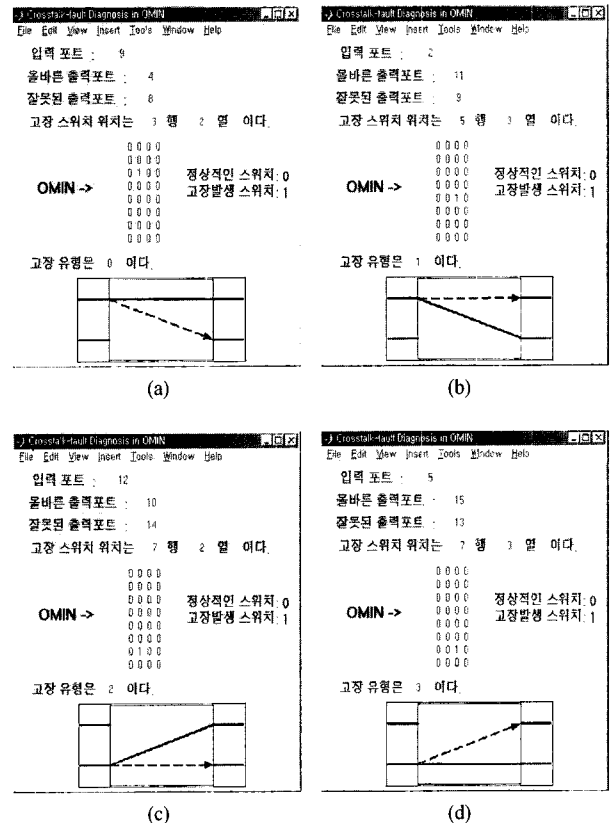


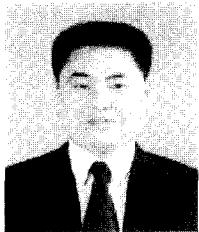
그림 12. Matlab(Ver. 6.0)을 이용한 모의실험 결과.
Fig. 12. Simulation results based on Matlab(Ver. 6.0).

은 각 단계에 사용된 스위치를 나타낸 것이다. 여기서 정상적인 스위치는 0으로 표기하고 누화고장이 발생하면 1을 출력하도록 하였다. 이를 통해 고장이 발생한 스위치의 위치 및 고장유형을 시각적으로 확인할 수 있다.

IX. 결론

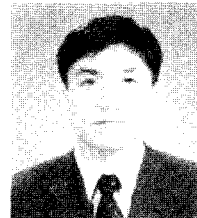
시스템에서 고장의 조기진단은 고장에 대한 신속한 대처를 가능하게 해주므로 시스템의 성능저하를 줄일 수 있을 뿐만 아니라, 시스템의 신뢰성을 높여주므로 매우 중요한

과제이다. 본 논문에서는 이러한 의미에서 OMIN에서의 고장진단기법에 대해서 고찰하였으며, 특히 OMIN에서 발생할 수 있는 누화고장에 대해 중점 연구하였다. 제시한 알고리즘은 기존의 테스트기법과는 달리 고장모델을 대수적으로 분석하여 해석적인 방법을 제시하였으며, 논리적 방법을 통한 빠른 연산으로 고장을 진단할 수 있었다. 또한, 현재 상용화된 2×2 결합기(coupler)를 이용한 스위치로 구성된 OMIN에 제시한 알고리즘을 적용하면 $O(\log m)$ 의 복잡도를 보였다[6]. 그리고 OMIN의 확장시 VRT만 새로이 구성하면 알고리즘의 변화 없이도 고장진단이 가능하므로 이를 통해 알고리즘적응의 효율성을 높일 수 있었다. 향후 연구과제로는 현재의 알고리즘을 수정 보완하여 다중 고장 발생시 이를 효과적으로 진단하기 위한 기법의 모색 등이 남아 있다.



김 영 재

2001년 울산대학교 전기공학과 졸업. 2001년~현재 울산대학교 전기전자정보시스템공학부 석사과정. 관심분야는 이산사건시스템의 관리제어, 광통신망 시스템(스위칭 및 교환기 분석), OXC 기술, 광 패킷교환 및 전송.



조 광 현

1993년 한국과학기술원 전기및전자공학과 졸업. 동대학원 석사(1995). 동대학원 박사(1998). 1998~1999 동대학원 위촉연구원. 1999. 3~현재 울산대학교 전기전자정보시스템공학부 조교수. 관심분야는 이산사건시스템의 해석 및 관리제어, 통신망 분석 및 제어, 광통신망 시스템(스위칭 및 라우팅), 시스템이론의 바이오정보공학 응용 등.

참고문헌

- [1] C.-L. Wu and T.-Y. Feng, "On class of multistage interconnection networks", *IEEE Trans. on computers*, vol. C-29, no. 8, 1980.
- [2] T. E. Stern and K. Bala, *Multiwavelength Optical Networks: A Layered Approach*, MA: Addison-Wesley, 1999.
- [3] N. K. Ailawadi, R. C. Alferness, G. D. Bergland, and R. A. Thompson, "Broadband photonic switching using guided-wave fabrics," *IEEE LTS Magazine*, vol. 2, no. 2, pp. 38-43, 1991.
- [4] Y. Pan, C. Qiao, and Y. Yang, "Optical multistage interconnection networks: New challenges and approaches," *IEEE Communications Magazine*, vol. 37, no. 2, pp. 50-56, 1999.
- [5] C. Qiao, "A two-level process for diagnosing crosstalk in photonic dilated benes networks," *Journal of Parallel and Distributed Computing*, vol.41, no.1, pp.53-66, 1997.
- [6] I.-S. Hwang and S.-N. Lee, "Diagnosing crosstalk faults in a class of dilated blocking optical multistage interconnection networks," *Journal of Information Sciences*, vol. 124, no. 1, pp. 59-91, 2000.
- [7] J. P. Hayes, *Computer Architecture and Organization*, NY: McGraw-Hill, 1988.
- [8] C.-S. Wu, G.-K. Ma, and B.-S. Lin, "Extended baseline architecture for nonblocking photonic switching," *Journal of Lightwave Technology*, vol. 15, no. 5, pp. 771-778, 1997.