

다중컴퓨터 시스템에서의 Beta-network의 링크선에 관한 Fault-tolerance 분석

(Fault-tolerance Analysis of Link Line of Beta-network in the Multicomputer System)

田 祐 千* 金 聖 天**

(Woo Chun Jun and Sung Chun Kim)

要 約

본 논문은 다중컴퓨터 시스템(multicomputer system)에서 사용되는 상호연결 네트워크(interconnection network)의 일종인 B-net(Beta-network)의 fault-tolerance에 대해서 고찰하였다.

본 논문에서는 임의의 B-net에 대하여 스위칭 소자(switching element) 사이를 연결하는 링크선(link line)의 maximal tolerable fault set(MTFS)를 구하는 방법을 제시하였다. 또 이를 이용하여 링크선에 오류(fault)가 발생했을 경우 DFA(dynamic full access) 성질을 확인할 수 있게 하고, B-net의 fault-tolerance의 정도를 결정하는 기준을 소개하였다.

Abstract

This thesis is concerned with fault-tolerance of a B-net (Beta-network) which is a kind of interconnection network in the multicomputer system.

In this paper, a method for obtaining Maximal Tolerable Fault Set (MTFS) of link line connecting switching elements in the arbitrary B-net is presented. Using this method, it is seen that testing of DFA capability is possible when s-a-faults of link line occur, and criterion for determining degree of fault-tolerance of a B-net in terms of link line is introduced.

I. 서 론

근래에 프로세서들을 서로 연결하여 계산능력을 증대시키려는 병렬처리(parallel processing)에 대한 연구가 활발히 진행되었다. 이들을 서로 연결하여 사용하는 목적은 병렬처리를 함으로써 계산속도를 향상시킬 수 있을 뿐 아니라 한 프로세서가 고장나더라도 나머지 것들을 계속 가동시킴으로써 신뢰도를 높일 수

있기 때문이다. 이것은 단일 프로세서가 처리할 수 있는 양보다 훨씬 더 많은 일을 처리할 수 있고, 또 VLSI 기술의 발달로 집적도가 높아지고, 메모리 등의 여러 소자의 가격이 하락함에 따라 더욱 필요성이 커졌다.

이러한 병렬처리 시스템의 일종인 다중컴퓨터 시스템은 INN(interconnection and communication network)에 의해 연결되는 계산장치(computing unit)들로 구성된 시스템이다. 각 계산장치는 데이터의 처리 및 저장, ICN에 대한 접속능력을 가지고 있고, ICN은 계산장치들 사이의 통신을 위해 연결선을 제공한다. ICN으로 사용될 수 있는 것은 공유버스(shared-bus), 전용 링크(dedicated link), 상호연결 네트워크 등 크

*準會員, **正會員, 西江大學校 電子計算學科
(Dept. of Computer Science, Sogang Univ.)
接受日字: 1987年 3月 2日

계 3 가지이다. 이들 중에서 상호연결 네트워크는 공유버스나 전용 링크에 비해서 여러가지 장점이 있기 때문에 널리 쓰인다.

일반적으로 $N \times N$ 상호연결 네트워크는 N 개의 입력 터미널과 N 개의 출력 터미널 사이의 연결선을 제공한다. 다중컴퓨터 시스템에서의 계산장치는 입력 터미널과 출력 터미널을 따로 구분하지 않고 그림 1 과 같이 동시에 입출력 터미널로서 고려된다. 여기에 사용된 B-net는 다중컴퓨터 시스템에서 널리 사용되는 상호연결 네트워크의 일종으로서 b-소자(beta-element)라 불리는 2×2 크로스바(crossbar) 스위치들로 구성되어 있다.¹¹⁾

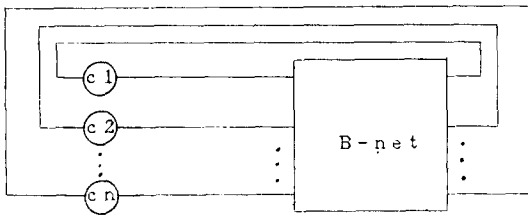


그림 1. B-net를 사용한 다중컴퓨터 시스템
Fig. 1. Multicomputer System Using a B-net.

그림 1 에서와 같이 B-net을 사용한 다중컴퓨터 시스템에서의 상호연결은 B-net의 출력에서 계산장치로의 “귀환선”(feedback path)을 통하여 가능하다. 이 귀환선은 계산장치 사이의 연결을 위하여 B-net을 통한 다중 거침(multiple pass)을 기본적으로 허용하고 있다. 이때 임의의 계산장치가 또 다른 계산장치로 B-net을 통한 유한 횟수의 다중거침을 거쳐 연결되어 full connection을 유지하였을 때 DFA(dynamic full access)의 성질을 갖고 있다 한다.¹²⁾

정상적으로, b-소자는 제어선(control line)에 주어진 논리값에 따라 두개의 가능한 상태(state)를 갖는다. 그림 2는 “through(T)”와 “cross(X)”의 서로 다른 두 연결상태를 보여 주고 있다.

본 논문은 B-net의 연결기능에 대해, b-소자의 오류가 미치는 영향을 중점 연구하였다. 일반적으로 IC(integrated circuit) 칩 내부의 논리적 오류보다는 핀(pin)에서 대부분 오류가 발생한다. 따라서 스위칭 소자(switching element)에 연결된 입력선, 출력선, 제어선의 오류를 고려하는 것이 중요하다.¹³⁾

b-소자에 연결된 입력선, 출력선, 제어선의 오류는 대부분 s-a-fault(stuck-at-fault)이다.¹⁴⁾ 한 b-소자의 링크선 즉 입력선과 출력선에 흐르는 논리값이 0 또는 1에 관계없이 항상 0(1)로 고정된 논리값을

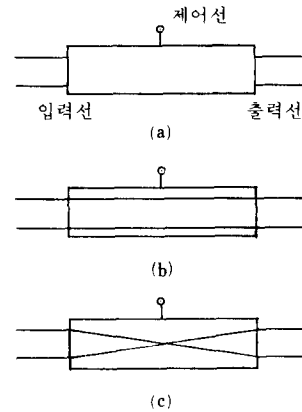


그림 2. b-소자의 연결상태
(a) 연결안된 상태
(b) T-상태 (c) X-상태

Fig. 2. Connection States of a b-element.
(a) Unconnected State.
(b) T-State. (c) X-State.

나타낼 때 링크선의 s-a-0(1) 오류라 한다.¹⁵⁾ 또한 b-소자에 연결된 제어선에 s-a-0(1) 오류가 발생하여 b-소자가 두 상태 중의 하나에 영구 고정되어 있을 때 제어선의 s-a-fault로 간주한다. ‘T’ 상태로 고정되었을 때는 s-a-T(stuck-at-through) 오류라 하며, ‘X’ 상태로 고정되었을 때는 s-a-X(stuck-at-cross) 오류라 정의한다.¹⁴⁾ 이처럼 오류는 스위칭 소자 자체의 s-a-fault(제어선의 s-a-fault) 및 링크선의 s-a-fault로 복합적으로 발생한다. s-a-fault들은 그것이 B-net의 DFA 성질을 파괴할 때 critical 오류라 하고 그렇지 않을 때를 noncritical 오류라 한다.¹²⁾

II. 연구 배경

지금까지 다중컴퓨터 시스템에서 사용되는 B-net의 fault-tolerance 연구는 대부분이 스위칭 소자에 연결된 제어선의 s-a-fault에 대해 중점적으로 진행되어 왔다.^{1, 6-10)}

B-net에서의 스위칭 소자의 제어선에 관한 fault-tolerance의 기준으로 fault-tolerance 변수 k 가 사용되었다. 즉 임의의 B-net가 n 개의 b-소자를 가질 때, 이 네트워크가 k -fault-tolerant(k -FT)를 가지는 것은 k 개 이하의 임의의 b-소자의 s-a-fault가 DFA 성질을 파괴하지 않는 것을 의미한다. 이를 만족하는 k 중에서 가장 큰 값이 B-net의 fault-tolerance 변수이다.¹²⁾

그러나 임의의 B-net에 대하여 스위칭 소자 사이를 연결하는 링크선, 즉 입력선과 출력선의 fault-tolerance에 관한 연구는 거의 없었다. 단지 스위칭 소자의

단일 오류(제어선 또는 링크선)에 대해 DFA 성질을 만족하는 가장 fault-tolerance가 큰 B-net가 제시되었다.¹¹⁾

본 논문에서는 링크선의 fault-tolerance에 대해서 중점적으로 고찰하였다. B-net의 링크선에 대한 MTFs(maximal tolerable fault set)를 구하는 방법을 제시함으로써 B-net의 fault-tolerance의 정도(degree)를 나타내는 척도로 활용될 수 있음을 보여 주고 아울러 링크선에 s-a-fault가 발생했을 때 DFA 성질을 확인할 수 있음을 보여 주고 있다.

III. MTFs 조사

본 논문은 B-net의 링크선에 대한 fault-tolerance를 조사하기 위해 다음과 같은 가정과 정의를 이용했다.

가정 : B-net의 링크선에 관한 fault-tolerance 분석을 위하여, s-a-fault는 중간 링크선에서만 발생하고 b-소자에 연결된 제어선의 s-a-fault는 발생하지 않는다고 가정한다.

정의 1 : B-net의 중간 링크선에 대한 MTFs는 중간 링크선의 s-a-fault에도 불구하고 DFA 성질을 만족하는 서로 독립된 최대의 오류 집합으로 정의한다.

즉, 그림 3 (a)에서와 같이 B-net의 링크선에 대한 s-a-fault는 기본 입력선과 기본 출력선을 제외한 중간 링크선에서만 발생한다고 본다. 이는 B-net의 기본 입력선과 기본 출력선에 s-a-fault가 발생하면 DFA 성질이 파괴되기 때문이다.¹¹⁾ 따라서 fault-tolerance를 얻지 못한다.

일반적으로 B-net가 N개의 입력선(출력선)과 m-단계로 구성될 경우에는 중간 링크선의 총 수는 $C = N \times (m-1)$ 이 됨을 알 수 있다.

B-net의 일종인 SSIN(single stage interconnection network)은 중간 링크선이 없으므로 본 논문에서의 B-net는 단지 MIN(multistage interconnection network)만을 의미한다.

B-net의 중간 링크선에 대한 MTFs를 구하기 위해서 다음과 같은 과정이 요구된다.

단계 1 : link fault identification procedure

B-net의 각 중간 링크선에 고유 링크선 번호를 부여한다.

```
for each intermediate link line
  ORGNUMBER (i) = i
end
// ORGNUMBER (i) : 각 중간 링크선의 고유 링크선 번호 //
```

단계 2 : DFA test procedure

중간 링크선의 가장 낮은 번호에서 시작하여 차례로 그 중간 링크선의 s-a-fault에도 불구하고 DFA 성질이 유지되면 그 보다 번호가 큰 중간 링크선들의 s-a-fault를 가정하여 이들과 조합하여 DFA 성질을 만족하는가를 조사한다.

procedure CHECKDFA

```
// 링크선 ORGNUMBER (i)의 s-a-fault를 가정하여, 이 링크선과 ORGNUMBER (i) 보다 큰 번호를 가진 링크선과 조합하여 DFA를 확인하는 알고리즘 //
```

for i ← 1 to C

// C : 중간 링크선의 총 갯수 //

if s-a-fault of link line of ORGNUMBER (i) is noncritical,

then call COMBI (i)

else

print, "s-a-fault of link line of ORGNUMBER (i) is critical"

end if

end

end CHECKDFA

procedure COMBI (i)

// 링크선 ORGNUMBER (i)와 조합하여, DFA를 만족하는 링크선을 찾는 알고리즘 //

j ← i+1

for k ← j to C

if s-a-fault of link line of ORGNUMBER (k) is noncritical,

then call COMBI (k)

end if

end

return

end COMBI

위의 과정을 적용하기 위하여 그림 3의 B-net의 링크선(기본 입력선, 기본 출력선, 중간 링크선)을 정점으로, b-소자에서 제공하는 연결선을 간선으로 하는 그래프 모델(그림 4 (a))을 고려한다.¹¹⁾ 이후의 링크선은 중간 링크선을 의미한다.

이 그래프 모델은 링크선 1에 s-a-fault가 발생했을 때 그림 4 (b)와 같게 된다. 이 그래프 모델을 기본 입력선과 기본 출력선 사이의 연결상태를 나타내는 방향 그래프(그림5)로 표현하여 강력 결합 상태인 가를 조사한다. 즉 한 방향 그래프가 강력 결합 상태이면,

그 방향 그래프 내의 임의의 정점들 사이에 경로(path)가 존재함을 의미한다. 이때 방향 그래프가 강력 결합 상태이면 DFA 성질이 만족되고, 아니면 DFA 성질이 파괴된 것이다.¹²⁾ 따라서 단계 2에서 보여주듯이 이 방향 그래프가 강력 결합 상태이면 다음 링크선의 s-a-fault를 가정하여 이와 조합하여 DFA 성질을 계속 확인한다.

그림 5의 방향 그래프는 강력 결합 상태를 만족하므로 링크선 2의 s-a-fault를 가정한 그래프 모델을 구성하여 DFA 성질이 만족되는가를 계속 확인한다. 이의 결과는 그림 6(a)에 나타나 있다.

즉, 그림 6(b)의 방향 그래프는 강력 결합 상태가

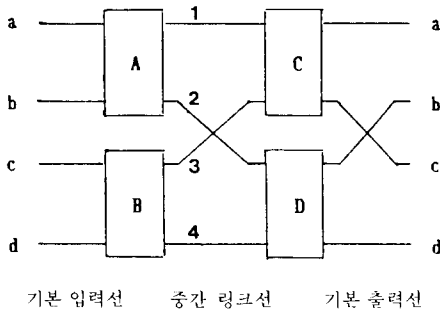


그림 3. Indirect Binary 2-cube 네트워크
Fig. 3. Indirect Binary 2-cube network.

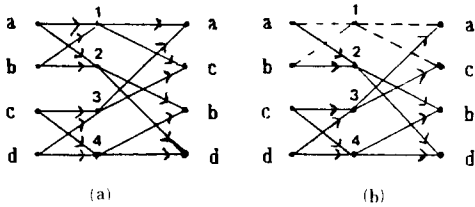


그림 4. 그림 3의 그래프 모델
(a) 그림 3의 그래프 모델
(b) 링크선 1을 없앴을 경우의 (a)의 그래프 모델

Fig. 4. Graph Model of Fig. 3.
(a) Graph Model of Fig. 3.
(b) Graph Model of (a) when Link Line 1 is Removed.

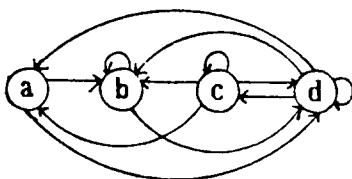


그림 5. 그림 4(b)의 방향 그래프
Fig. 5. Directed Graph of Fig.4(b).

아님을 보여 준다. 그러므로 링크선 1과 2에 동시에 s-a-fault가 발생하면 DFA 성질을 만족하지 못하는 critical 오류임을 확인한다.

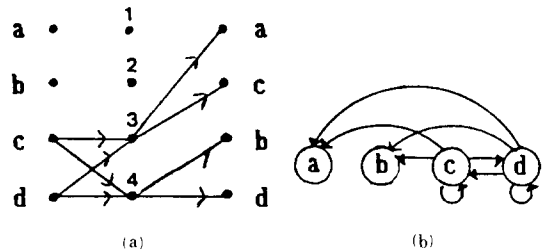


그림 6. 그림 3의 그래프 모델과 방향 그래프
(a) 그래프 모델 (b) (a)의 방향 그래프

Fig. 6. A Graph Model and its Directed Graph of Fig.3.
(a) Graph Model. (b) Directed Graph of (a).

IV. 링크선 조합 트리의 구성

본 장에서는 MTFs를 구하기 위한 단계 2 (DFA test procedure)를 구체화하기 위한 링크선 조합 트리를 제시한다.

단계 2에서 모든 링크선들의 s-a-fault를 가정할 때 각각의 링크선에 대한 DFA 성질을 조사하는 방법은 각 링크선에 대하여 조합할 수 있는 모든 링크선을 전부 검토하는 것이다. 이를 구체화하기 위해 다음을 정의한다.

정의 2 : 단계 1이 수행된 후 B-net에 대한 고유번호가 가장 낮은 중간 링크선에서 시작하여 차례로 모든 중간 링크선에 대해서, 각각 그 보다 번호가 큰 중간 링크선과의 모든 조합을 이루는 c개 (c : B-net의 중간 링크선의 총 갯수)의 트리로 구성된 트리를 링크선 조합 트리라 정의한다. 이 링크선 조합 트리에서 각 노드는 단계 1에서 고유번호가 부여된 B-net의 중간 링크선을 나타낸다.

링크선 조합 트리를 각각의 루트로부터 탐색할 때, 그 노드가 나타내는 중간 링크선들의 s-a-fault에도 불구하고 DFA 성질이 만족되면 다음 레벨에 있는 노드들이 나타내는 중간 링크선들의 s-a-fault를 가정하여 이들과 결합하여 DFA 성질을 확인할 수 있다.

앞서 정의한 링크선 조합 트리가 모든 중간 링크선들의 조합을 포함하는가의 여부는 다음 정리의 결과로 알 수 있다.

정리 1 : 링크선 조합 트리는 링크선들의 모든 조합을 포함한다.

(증명) B-net의 모든 중간 링크선의 갯수가 c일때,

이 링크선들의 모든 조합은 c 개의 각기 다른 링크선들로 구성된 조합, $c-1$ 개의 각기 다른 링크선들로 구성된 조합, ..., 1개의 링크선들로 구성된 조합으로 각각 분류할 수 있다.

링크선 조합 트리에서 링크선 1에 대한 트리는 링크선 1부터 c 까지의 모든 링크선들의 조합을 다 포함한다. 링크선 2에 대한 트리는 링크선 2부터 c 까지의 모든 링크선 조합을 다 포함한다. 마찬가지로 링크선 c 에 대한 트리는 자신만을 포함하는 링크선 조합이다. 링크선 1부터 차례로 조사할 때, 링크선 1에 대한 트리를 구하면 원소의 갯수가 c 인 모든 조합, 즉 c 개의 각기 다른 링크선으로 구성된 모든 조합이 다 조사되고, 또 링크선 2에 대한 트리를 구하면 원소의 갯수가 $c-1$ 인 모든 조합이 조사된다. 마찬가지로 링크선 c 에 대한 트리를 구하면 원소의 갯수가 1인 모든 조합이 조사된다. 따라서 링크선 1에서 c 까지의 트리는 즉 링크선 조합 트리는 링크선들의 모든 조합을 포함한다(Q. E. D.).

그 예로 그림 3의 B-net의 경우 모든 중간 링크선들의 s-a-fault를 가정하면, 링크선들의 모든 조합은 그림 7과 같이 트리의 형태로 나타낼 수 있다.

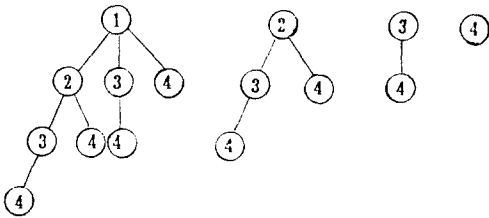


그림 7. 그림 3의 링크선 조합트리
Fig. 7. Link Line Combination Tree of Fig.3.

즉, 링크선 1의 s-a-fault에도 불구하고 DFA 성질이 만족되면 다음에 링크선 2, 3, 4와의 모든 조합에 대하여 계속 조사한다. 그러나 만일 링크선 1에 대하여 DFA 성질이 파괴되면 링크선 2부터 조사한다. (그림 7의 두번째 트리). 이와 같이 링크선 2, 3, 4에 대해서 모두 조사한다.

MTFS 조사의 단계 1에서 중간 링크선에 번호를 붙이는 것은 링크선들의 조사 순서가 DFA 성질에 영향을 미치지 않기 때문에 임의로 번호를 붙여도 된다. 즉, 중간 링크선에 부여하는 논리번호에 관계없이 이들의 링크선 조합 트리는 링크선들의 모든 조합을 다 포함하고, 또 가령 임의의 B-net의 두 링크선에 i 와 j 의 논리번호가 각각 부여되었을 때 이들의 s-a-fault에도 불구하고 DFA 성질이 만족되면($i \neq j$), j 와 i 로 논

리번호가 부여되어도 마찬가지로 DFA 성질이 만족된다. 링크선 조합 트리에 관한 정리 1로부터 다음과 같은 특성을 유도할 수 있다.

특성 1 : 임의의 B-net가 c 개의 중간 링크선들을 가질 때 링크선 조합 트리의 모든 노드들의 수는 $2^c - 1$ 이다.

링크선 조합 트리에서 레벨이 n 인 경우에 ($1 \leq n \leq c$), 이 레벨에서 노드들의 총 갯수, 즉 원소의 갯수가 n 인 조합의 총 수는 c 개에서 임의의 n 개를 선택하는 것과 마찬가지로 $\binom{c}{n}$ 이다. 따라서 각 레벨에서의 조합의 수는 각각 $\binom{c}{1}$, $\binom{c}{2}$, $\binom{c}{3}$, ..., $\binom{c}{c}$ 이다. 이들의 총합은 $\binom{c}{1} + \binom{c}{2} + \binom{c}{3} + \dots + \binom{c}{c} = 2^c - 1$ 이다.¹³⁾

V. MTFS의 생성

링크선 조합 트리의 주어진 모든 조합에서, DFA 성질을 만족시키지 못하는 조합은 아래와 같은 특성을 고려하면 간단히 추론해 낼 수가 있다.

특성 2 : B-net의 기본 입력선(또는 기본 출력선)에 연결된 b-소자는 이에 연결된 두 중간 링크선에 s-a-fault가 발생하면 DFA 성질이 파괴된다.

즉, B-net의 기본 입력선에 연결된 한 b-소자의 두 중간 링크선에 s-a-fault가 발생하면 기본 입력선은 B-net를 통해 데이터를 전혀 전달할 수 없고 마찬가지로 기본 출력선에 연결된 한 b-소자의 두 중간 링크선에 s-a-fault가 발생하면 기본 출력선은 데이터를 전혀 받을 수 없다.

예로서 그림 3에서 b-소자 A의 링크선 1, 2 또는 b-소자 B의 링크선 3, 4에 동시에 s-a-fault가 발생되면 기본 입력선 a, b 또는 c, d에서 전혀 데이터를 전달할 수 없고, 또한 b-소자 C의 링크선 1, 3 또는 b-소자 D의 링크선 2, 4에 동시에 s-a-fault가 발생되면 기본 출력선 a, c 또는 b, d에서 데이터를 받을 수 없기 때문에 DFA 성질이 파괴된다.

특성 2를 이용하여 링크선 조합 트리를 다음과 같이 간소화할 수 있다.

정의 3 : B-net에 대한 링크선 조합 트리에서 **특성 2**를 만족하는 임의의 조합 즉 기본 입력선(출력선)에 연결된 b-소자의 두 중간 링크선에 s-a-fault가 발생한 경우의 이들 두 중간 링크선의 조합을 (ℓ, m) (단, $\ell < m, 1 \leq \ell, m \leq c, c$ 는 B-net의 중간 링크선의 총 갯수)이라 할 때 m 이하의 모든 노드들을 제거시킨 링크선 조합 트리를 **간소화된 링크선 조합 트리**라 정의한다.

정리 2 : 간소화된 링크선 조합 트리는 **특성 2**를 만족하는 조합을 제외한 모든 조합을 다 포함한다.

(증명) c 개의 중간 링크선을 갖는 B-net의 링크선

조합 트리는 정의에 따라 1, 2, ..., c를 각각 루트 노드로 하는 c개의 트리로 구성되어 있다. 임의의 $i(1 \leq i \leq c)$ 를 루트로 하는 트리는 i에서 c까지의 모든 링크선들의 조합을 포함한다. 따라서 링크선 i와 c 사이에 특성 2를 만족하는 링크선들의 조합을 제거하여도 DFA 성질을 유지할 수 있는 가능한 모든 조합은 남아있기 때문에 간소화된 링크선 조합 트리는 모든 조합을 다 포함한다. 예를 들어 아래 그림에서와 같이, i를 루트로 하는 트리에서 조합 (i, i+1)이 특성 2를 만족하는 조합이라 하자. 노드 i+1 이하를 모두 제거하더라도 노드 i+2, ..., c는 노드 i에 계속 연결되어 있기 때문에 이 트리는 링크선 i와 i+2, ..., c의 조합을 포함한다(Q. E. D.).

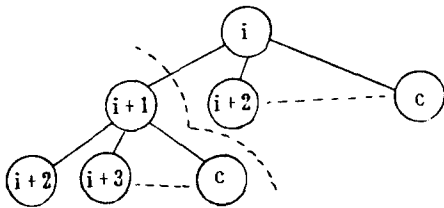


그림 3의 B-net에서 특성 2를 만족하는 조합은 (1, 2), (3, 4), (1, 3), (2, 4)이므로 이의 링크선 조합 트리(그림 7)는 그림 8과 같이 간소화된다.

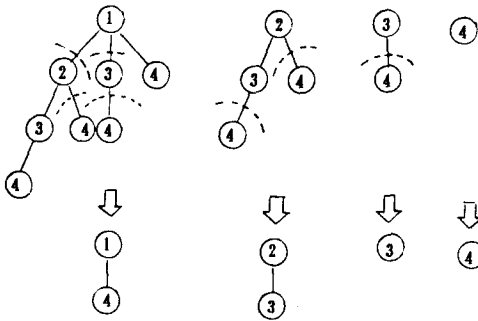


그림 8. 그림 7의 간소화된 링크선 조합트리
Fig. 8. Simplified Link Line Combination Tree of Fig. 7.

이 간소화된 링크선 조합 트리의 탐색을 통해서 얻어진, DFA 성질을 만족하는 링크선들의 서로 독립된 최대 집합이 MTFS이다. 그림 3의 B-net에서의 MTFS는 {1, 4}, {2, 3}이다.

정리 3 : 임의의 B-net의 중간 링크선들에 대한 논리번호 부여순서에 관계없이 간소화된 링크선 조합 트리의 노드들의 수는 일정하다.

(증명) c개의 중간 링크선을 갖는 B-net에서 링크선

조합 트리는 원소가 1개인 즉 각각의 서로 다른 링크선이 1개인 조합, 2개인 조합, ..., c개인 조합을 각각 포함한다. 각각의 조합의 수는 특성 1로부터 (1), (2), (3), ..., (c)이다. 특성 2를 만족하는 조합은 원소가 1개인 조합에는 영향을 미치지 않는다. 원소가 2개인 조합은 (2)에서 특성 2를 만족하는 조합을 제외시킨 것이다. 원소가 3개인 조합은 각각의 링크선이 특성 2를 만족하지 않는 조합으로 구성된다. 마찬가지로 원소가 c개인 조합도 각각의 링크선이 특성 2를 만족하지 않는 조합으로 구성된다. 이러한 각각의 조합은 링크선에 부여한 논리번호에 관계없이 그 수가 일정하므로 간소화된 링크선 조합 트리의 노드들의 수는 일정하다(Q. E. D.).

그림 3과 같이 논리번호를 부여했을 경우, 링크선의 갯수가 1인 조합은 (1), (2), (3), (4)가 된다. 링크선의 갯수가 2인 조합은 (1, 2), (3, 4), (1, 3), (2, 4)를 제외한 조합이므로 (1, 4), (2, 3)이 된다. 링크선의 갯수가 3인 조합은 (1, 2, 3), (1, 2, 4), (1, 3, 4), (2, 3, 4)로서, 이들은 모두 조합 (1, 2), (3, 4), (1, 3), (2, 4) 중의 하나 이상을 포함하므로 실질적인 조합은 없게 된다. 마찬가지로 링크선의 갯수가 4인 조합도 없다. 따라서 특성 2를 만족하는 모든 조합을 제외한 조합은 (1), (2), (3), (4), (1, 4), (2, 3)이 되고 이것은 그림 8과 같은 간소화된 링크선 조합 트리로 나타난다. 그림 9와 같이 논리번호가 부여되었을 경우는 특성 2를 만족하는 조합은 (1, 3), (1, 4), (2, 3), (2, 4)가 되어 특성 2를 만족하는 모든 조합을 제외한 조합은 (1), (2), (3), (4), (1, 2), (3, 4)로서 간소화된 링크선 조합 트리는 그림 10과 같다.

그림 8에서 알 수 있듯이 그림 3의 B-net에 대한 링크선 조합은 각각 (1, 4), (2, 3), (3), (4)로서 간소화된 링크선 조합 트리의 노드의 수는 6개이다. 그림 9에 대한 링크선 조합은 각각 (1, 2), (2), (3, 4), (4)이므로 마찬가지로 간소화된 링크선 조합 트리의 노드의 수는 6개이다.

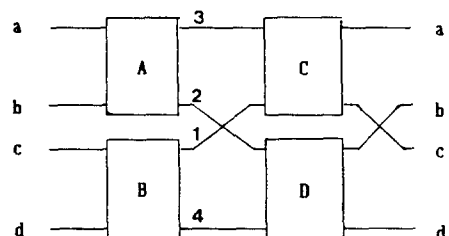


그림 9. 그림 3의 다른 논리번호 부여
Fig. 9. Different Logical Number Ordering of Fig. 3.

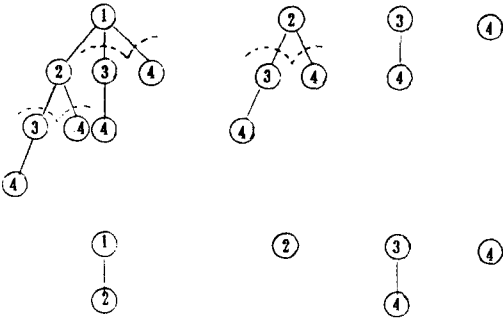


그림10. 그림 9의 간소화된 링크선 조합트리
Fig.10. Simplified Link Line Combination Tree of Fig. 9.

VI. MTFFS의 응용

본 장에서는 임의의 B-net에 대한 MTFFS의 응용을 제시한다. 이를 위해 다음과 같은 표기를 이용한다.

표기: 임의의 $N \times N$ 입력선/출력선 m -단계 B-net (MIN)을 가정하여, 중간 링크선들이 특성2를 만족하는 경우의 조합을 제외한 모든 조합에 대해, 임의의 n 개의 s -a-fault가 발생했을 때 DFA 성질을 유지할 수 있는 최대 n 을 $C(N, m) = n$ 이라 나타낸다.

이 $C(N, m)$ 을 B-net의 중간 링크선들에 대한 fault-tolerance의 기준으로 이용한다. 그 예로 4×4 2-단계 B-net(baseline 네트워크)와 이의 간소화된 링크선 조

합 트리가 그림11에 있다.

이때 B-net의 MTFFS는 $\{1, 4\}$ 이다. 특성2를 만족하는 조합을 제외한 모든 링크선 조합 $(1, 4), (2, 3), (3), (4)$ 에 대해 링크선 2이나 3의 s -a-fault는 critical이므로 이 B-net에 대해 $C(4, 2) = 0$ 이라 할 수 있다. 이에 비해 그림3에서의 B-net의 특성2를 만족하는 조합을 제외한 모든 링크선 조합은 그림11의 B-net과 동일하나 MTFFS는 $\{1, 4\}, \{2, 3\}$ 이므로 임의의 2개의 링크선의 s -a-fault에 대해 noncritical 이어서 $C(4, 2) = 2$ 라 할 수 있다. 그림3의 indirect binary 2 cube 네트워크와 그림11의 baseline 네트워크는 topology가 같음이 입증되어 있다.^[14] 이처럼 $C(N, m)$ 은 네트워크의 topology가 같더라도 기본 입력선과 기본 출력선의 링크 패턴에 따라 그 값이 변할 수 있다.

또 B-net의 MTFFS를 이용하여 중간 링크선에 s -a-fault가 발생했을 때 DFA 성질을 확인할 수 있다. 즉 중간 링크선의 s -a-fault들이 MTFFS의 부분집합이면 이 s -a-fault는 noncritical이고, s -a-fault들이 MTFFS의 어느 부분집합에도 속하지 않으면 DFA 성질을 파괴하는 critical 오류라는 것을 확인할 수 있다.

예로서 그림3의 B-net에서 링크선 1과 3에 s -a-fault가 발생했다고 가정하자. 이 B-net의 MTFFS는 $\{1, 4\}, \{2, 3\}$ 이므로 링크선 1과 3의 s -a-fault는 critical이 된다. 그러나 링크선 4의 s -a-fault를 가정하면 이 오류는 $\{1, 4\}$ 의 부분집합이므로 noncritical이 된다.

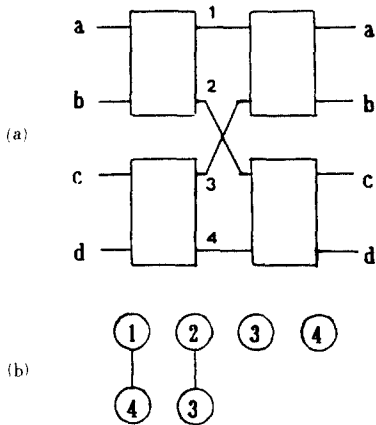


그림11. 4×4 2-단계 B-net와 간소화된 링크선 조합 트리

- (a) 4×4 2-단계 B-net
- (b) (a)의 간소화된 링크선 조합트리

Fig. 11. 4×4 2-Stage B-net and its Simplified Link Line Combination Tree.

- (a) 4×4 2-Stage B-net.
- (b) Simplified Link Line Combination Tree of (a).

VII. 결론 및 연구방향

본 논문은 다중 컴퓨터 시스템에서 사용되는 B-net의 fault-tolerance에 대해서 고찰하였다. 즉 B-net에서 발생할 수 있는 제어선과 링크선의 s -a-fault 중에서 링크선의 s -a-fault에 대해 중점 연구하였다.

본 논문에서는 B-net에서의 링크선의 s -a-fault에 대한 MTFFS를 구하는 방법을 제시하였다. MTFFS를 이용하여 임의의 B-net의 링크선에 대한 fault-tolerance의 정도를 설정하는 기준을 제시하였고, 또한 링크선의 s -a-fault에 대하여 DFA 확인 방법으로 이용될 수 있음을 보였다.

본 논문에서 앞으로 더 연구되어야 할 분야는 다음과 같다.

- MTFFS를 구하는데 있어서의 효율적인 simulation
- 여러 B-net에 대한 링크선의 fault-tolerance 비교 분석
- 링크선의 s -a-fault에 대한 fault-tolerance가 가장 좋은 B-net의 고안

參 考 文 獻

- [1] D.P. Agrawal, "Graph theoretical analysis and design of multistage interconnection networks", *IEEE trans. on computer*, c-32, no.7, pp.637-648, July 1983.
- [2] J.P. Shen and J.P. Hayes, "Fault tolerance of dynamic full-access interconnection networks", *IEEE Trans. on Computer*, c-33 no.3, pp.241-248, March 1984.
- [3] S.C. Kim and D.P. Agrawal, "Easily testable interconnection network from three-state cells", *16th Int. Conf. on Sys. Science*, pp.234-243, 1983.
- [4] D.P. Agrawal, "Testing and fault-tolerance of multistage interconnection networks", *IEEE Computer*, 15, pp.41-53, April 1982.
- [5] M.A. Breuer and A.D. Friedman, *Diagnosis and Reliable Design of Digital Systems*, Computer Science Press, Woodland Hills, pp.15-18, 1976.
- [6] J.P. Shen and J.P. Hayes, "Fault tolerance and performance analysis of beta-networks", *Parallel Computing*, 3, no.3, pp. 231-249, July 1986.
- [7] L. Climiniera and A. Serra, "Performance-fault tolerance tradeoffs in single stage connecting networks", *FTFS. 12th. Annual International Symposium*, pp.209-212, 1982.
- [8] J.P. Shen and J.P. Hayes, "Synthesis of fault-tolerant beta-networks", *FTFS. 12th. Annual International Symposium*, pp.201-208, 1982.
- [9] J.P. Shen and J.P. Hayes, "Fault tolerance of a class of connecting networks", *7th Symposium on Computer Architecture*, pp.61-71, May 1980.
- [10] J.P. Shen, *Fault Tolerance of Beta-Networks in the Interconnected Multicomputer Systems*, Ph.D. Dissertation, Dep. of Elec. Eng., Univ. of Southern Calif., Aug. 1981.
- [11] D.P. Agrawal and J.S. Leu, "Dynamic accessibility testing and path length optimization of multistage interconnection networks", *IEEE Trans. on Computer*, C-34, 255-266, March 1985.
- [12] 전우천, 김성천, "Digraph의 특성을 활용한 Beta-network의 dynamic full accessibility에 관한 fault tolerance 분석", 대한전자공학회 전자계산연구회 학술발표논문집, vol. 5, no. 2, pp. 108 - 114, 1986.
- [13] Alan Tucker, *Applied Combinatorics*, John Wiley and Sons, New York, pp.60-67, 1980.
- [14] D.P. Agrawal and S.C. Kim, "On Equivalent multistage interconnection networks", *Proc. 1981 Int. Conf. Parallel Processing*, pp.234-257, Aug 1981.