큐브 부호화 방식을 사용하는 다단계 상호연결망 기반의 다중컴퓨터에서 고장 허용 멀티캐스팅

김 진 수[†] · 박 재 형[††] · 김 명 균[†††]

요 약

본 논문에서는 다단계 상호연결망(MIN)을 기반으로 하는 다중컴퓨터에서의 고장 허용 멀티캐스팅에 대해 연구한다. 프로세싱 노드간 일대일 라우팅뿐만 아니라, 효율적인 멀티캐스팅은 다중컴퓨터의 성능에 영향을 주는 중요한 요소이다. 본 논문은 스위칭 소자의 고장을 허용하는 멀티캐스팅 알고리즘을 제안한다. 제안된 알고리즘은 멀티캐스트 목적지들을 표현하기 위해 큐브 부호화 방식을 사용하며, 고장을 우회하기 위해 순환 기법을 기반으로 한다. 이 알고리즘은 다수의 고장 스위칭 소자가 있는 MIN에서 임의의 멀티캐스트 메시지를 두 번 순환시켜 라우팅할 수 있다. 그리고, MIN의 구조적 특성을 이용하여 본 알고리즘의 정확성을 증명한다.

Fault-Tolerant Multicasting in MIN-based Multicomputers using Cube Encoding Scheme

Jinsoo Kim[†] · Jaehyung Park[††] · Myung-Kyun Kim[†††]

ABSTRACT

In this paper, we study fault-tolerant multicasting in multicomputers based on multistage interconnection networks (MIN's). In addition to one-to-one routing among processing nodes, efficient multicasting has an important effect on the performance of multicomputers. This paper presents a multicasting algorithm to tolerate faulty switching elements. The proposed algorithm uses the cube encoding scheme to represent multicast destinations in MIN, and is based on a recursive scheme in order to bypass faults. This algorithm can route any multicast message to its own destinations in only two passes through the MIN containing several faulty switching elements. Moreover, we prove the correctness of our algorithm by exploiting well-known nonblocking property of MIN.

키워드 : 다단계 상호연결망(mutistage interannection network), 큐브 부호화(cube encoding), 고장 스위칭 소자(faulty switching element), 멀티캐스팅(multi casting), 다중컴퓨터(multicomputer)

1. 서 론

다중컴퓨터는 대규모 병렬처리 작업을 수행하기 위한 수단으로 널리 사용되고 있다. 이 다중컴퓨터 시스템은 특정한 형태로 연결된 다수의 프로세싱 노드들로 구성되고, 각 노드는 프로세서, 지역 메모리, 통신장치 등을 갖고 있으며, 이 통신장치를 통해 노드간의 정보를 교환한다. 다중컴퓨터 시스템에서, 정보 교환의 형태는 일대일 통신(unicast communication) 뿐아니라 노드 그룹 간의 통신인 집합체 통신(collective communication) 기능의 중요성이 증가되고 있다[1, 2]. 멀티캐스트 통신은 집합체 통신의 가장 중요한 형태 가운데 하나로서, 하나의 출발지에서 임의의 다수 목적지로 동일한 메시지를 전송하는 일대다 통신을 의미한다. 이 멀티캐스트 통신은 병렬 처리에서 제공되는 경계선 동기화(barrier synchronization)와 같은 연산에 필수적이며, 또한 분산 공유 메모리 시스템에서 캐쉬의 일관성 유지 프로토콜을 위한 수정(updating)과 무효화(invalidation)에도 효과적으로 사용된다.

다단계 상호연결망(Multistage Interconnection Network, MIN)은 대규모 다중컴퓨터의 구성을 위해 사용되는 일반적이고 효율적인 연결망 구조중 하나이며, IBM SP1/SP2[3]과 NEC Cenju-3[4] 등에서 사용되고 있다. MIN은 임의

※ 본 연구는 한국과학기술원의 해킹바이러스 연구센터(Hacking and Virus Research Center)의 지원을 받았습니다.
† 정 회 원 : 건국대학교 컴퓨터응용학부 교수
†† 정 회 원 : ETRI 네트워크기술연구소 선임연구원
††† 정 회 원 : 울산대학교 컴퓨터정보통신공학부 교수
논문접수 : 2001년 2월 21일, 심사완료 : 2001년 5월 23일

의 입력단에서 모든 출력단으로 네트워크를 한번 통과하여 접근이 가능한 완전 접근 능력(full access capability)을 갖고 있다. 또한, MIN의 입력단과 출력단 사이에는 오직 하나의 라우팅 경로만이 존재하며, 이 특성으로 MIN에서 빠르고 효율적인 라우팅을 수행할 수 있다. 그러나 이 특성 때문에 MIN은 다른 네트워크에 비해 고장에 취약하다. 즉 MIN에서 스위칭 소자(Switching Element, SE) 또는 링크가 고장나면, 우회 경로가 없기 때문에 완전 접근 능력을 상실하게 된다.

MIN 기반의 다중컴퓨터에서 고장 허용성(fault-tolerance)의 실현은 크게 하드웨어 접근 방법과 소프트웨어 접근 방법으로 구분된다. 하드웨어 접근 방법은 네트워크에 SE 또는 링크를 추가하여 다중의 경로를 제공함으로써 고장을 극복하는 것이다[5, 6]. 이 방법에서는 고장이 발생하면 네트워크를 재구성하여 완전 접근 능력을 유지한다. 그러나, 고장 처리를 위해 추가되는 하드웨어 자원은 고장이 없으면 상당한 낭비가 되고, 네트워크를 불규칙한 형태로 만들어 다중컴퓨터의 모듈화를 저해하게 된다. 소프트웨어 접근 방법은 부수적인 하드웨어 없이 고장 허용 라우팅을 제공하는 것이다[7-9]. 이 방법에서는 메시지를 네트워크를 통해 여러 번 순환시킴으로써, 고장난 SE나 링크를 우회하도록 한다. 네크워크 상에서 제한된 수의 순환을 거쳐서 임의의 입력단에서 모든 출력단으로 접근이 가능한 특성을 동적인(dynamic) 완전 접근 능력이라고 한다[7]. MIN에서 하나의 고장으로 인해 완전 접근 능력은 상실되지만, 다수의 고장이 있는 경우도 동적인 완전 접근 능력은 유지될 수 있다. MIN에서 중간 출력단을 경유하여 메시지를 순환시켜 라우팅하는 기법을 순환 기법(recursive scheme)이라고 하며 [10], 이 기법을 이용하여 자원의 낭비 없이 고장을 우회하여 출발지(source)에서 목적지(destination)로 메시지를 라우팅 할 수 있다. 그러나 MIN에서 고장 허용라우팅에 대한 연구들은 대부분 일대일 통신을 대상으로 진행되었다.

MIN에서 순환 기법은 일대일 고장 허용 라우팅뿐만 아니라 멀티캐스팅의 실현을 위해서도 사용되고 있다. 순환 기법을 기반으로 하여 많은 멀티캐스팅 알고리즘이 기존에 제안되었으나[10-14], 고장 허용성은 거의 고려되지 않았다. 또한, 전술한 바와 같이 고장에 취약한 MIN 구조에서 고장을 허용하는 연구들은 일대일 통신에 국한되어 왔다. 본 논문은 MIN 기반의 다중컴퓨터에서 다수의 고장 SE들을 허용하는 멀티캐스팅 알고리즘을 제안한다. 제안된 알고리즘은 임의의 멀티캐스트 메시지의 주소를 표시하기 위해 큐브 부호화 방식을 사용하고, 또한 고장을 우회하고 멀티캐스트 메시지를 목적지들로 보내기 위해 순환 기법을 기반으로 한다. 그리고, MIN 구조가 갖는 무충돌(nonblocking) 특성을 활용하여, 다수의 특정한 SE들이 고장난 MIN에서 메시지를 단지 두 번 순환시켜 멀티캐스팅 한다.

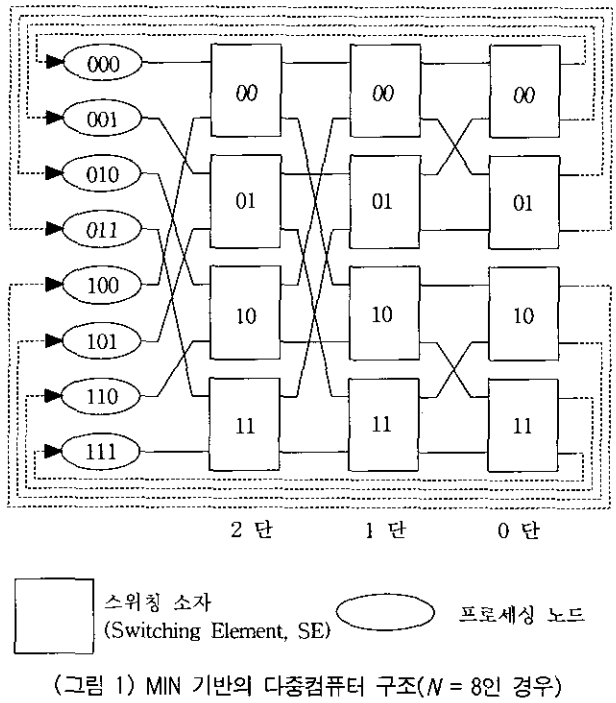
본 논문의 구성은 다음과 같다. 제2장에서는 다중컴퓨터 시스템과 MIN의 구조에 대해 기술하고, 또한 MIN에서의 멀티캐스트 주소 표현을 위한 큐브 부호화 방법과 MIN의 고장 모델, 그리고 본 논문에서 사용되는 용어들에 대해 설명한다. 제3장에서는 본 논문에서 제안하는 MIN에서의 고장허용 멀티캐스팅 알고리즘에 대해 서술하고 그 알고리즘의 정확성을 증명한다. 마지막으로 제4장에서 결론을 맺는다.

2. 시스템 모델과 용어

본 장에서는 MIN을 기반으로 하는 다중컴퓨터 시스템의 구조와 네트워크가 지닌 무충돌 특성에 대해 살펴보고, 멀티캐스트 주소 표현을 위한 큐브 부호화 방식에 대해 설명한다. 그리고, MIN에서의 고장 모델과 알고리즘에서 사용되는 기본적인 용어 등을 서술한다.

2.1 시스템 구조

네트워크 크기가 N인 MIN은 $n = log_2 N$ 개의 단(stage)으로 구성되고, 각 단은 $N/2$개의 2×2스위칭 소자(switching elements, SE)들로 이루어진다. 각 단에서 N개의 입출력 링크는 위부터 아래로 0에서 N-1까지 n비트로 표현되고, $N/2$개의 SE는 위부터 아래로 0에서 $N/2$-1까지 n-1비트로 표현된다. 각 단의 번호는 좌측부터 우측으로 n-1에서 0으로



(그림 1) MIN 기반의 다중컴퓨터 구조(N = 8인 경우)

표현된다.

본 논문에서 고려하는 MIN 기반의 다중컴퓨터는 (그림 1)과 같은 구조를 갖는다. (그림 1)에서 보는 바와 같이, 인접한 두 단간 버터플라이(butterfly) 연결 형태, 노드들과 $(n-1)$ 단 사이는 완전 셔플(perfect shuffle) 연결형태를 갖는다. 그리고 메시지의 순환을 위해, (그림 1)의 점선과 같이 우측의 출력단에서 좌측의 노드쪽으로 역방향 링크들이 존재한다. 또한, 멀티캐스트를 지원하기 위해, 모든 SE들은 브로드캐스트 기능을 갖는다. 즉 SE에서 입력 메시지를 한쪽 출력단으로 라우팅하는 기능을 수행하거나, 양쪽 출력단으로 브로드캐스팅하는 기능을 수행할 수 있다.

2.2 MIN의 구조적 특성

MIN은 2.1절에서 설명한 연결구조에 따라 다음과 같은 특성을 갖는다.

[특성-1] 입력이 $s = s_{n-1} \cdots s_1 s_0$, 출력이 $d = d_{n-1} \cdots d_1 d_0$인 메시지가 $i\,(n-2 \geq i \geq 1)$ 단을 통과할 때, 사용되는 SE 번호는 $(d_{n-1} \cdots d_{i+1} s_{i-1} \cdots s_0)$이며 그 SE의 d_i번째 출력 포트가 선택된다.

본 논문에서 고려하는 MIN은 오메가(omega) 네트워크와 동등한 구조적 특성을 갖고 있다[15]. 따라서 오메가 네트워크의 연구결과를 활용하여, 다음과 같은 두 메시지 사이의 무충돌 특성을 쉽게 유추할 수 있다 [16, 17].

[정의-1] $S_{msb}(a, b)$는 a와 b의 2진수 표현에서 서로 일치하는 최상위 비트들의 개수이며, $S_{lsb}(a, b)$는 일치하는 최하위 비트들의 개수이다.

예를 들어 $S_{msb}(0110, 0010)$은 1이고, $S_{lsb}(0110, 0010)$은 2이다.

[특성-2] 메시지 m의 입력과 출력을 각각 s^m과 d^m라고 표시할 때, 두 메시지 i와 j가 MIN에서 충돌이 발생되지 않는 필요충분 조건은 $S_{lsb}(s^i, s^j) + S_{msb}(d^i, d^j) < n$ 이다.

2.3 큐브 부호화 방식

멀티캐스트 목적지의 수는 일정하지 않다. 따라서, 멀티캐스트 메시지의 목적지들을 표현하는 여러 가지 효과적인 방식들이 사용되며, 다중컴퓨터에서 널리 사용되는 방식 중 하나가 큐브 부호화 방식이다[11-13]. 큐브 부호화 방식은 몇 개의 목적지를 하나의 큐브로 합쳐서 표현하는 방식이다. 예를 들어 멀티캐스트 목적지가 {010, 011, 100}일 때, 멀티캐스트 목적지 큐브는 01*와 100이 된다. 즉 하나의 큐브 01*는 두 개의 목적지 010과 011을 원소로 갖는 집합으로 볼 수 있다.

큐브 부호화 방식에서 멀티캐스트 라우팅 헤더는 $\{R, M\}$으로 표시되며, $R = r_{n-1} \cdots r_1 r_0$은 라우팅 정보를 포함하며 $M = m_{n-1} \cdots m_1 m_0$은 멀티캐스트 정보를 포함한다. 멀티캐스트 헤더 $\{R, M\}$을 처리하기 위해, $i\,(n-1 \geq i \geq 0)$ 단의 SE들은 r_i와 m_i를 조사한다. 만일 m_i가 0이면 r_i에 따라 일대일 라우팅을 수행하고, 만일 m_i가 1이면 r_i를 무시하고 브로드캐스팅을 수행한다.

[정의-2] 두 큐브 C_1과 C_2 간의 **대소 관계** $<_D$는 다음과 같이 정의한다.

- $C_1 <_D C_2$의 필요충분조건은, 각각 C_1과 C_2에 속한 모든 d^1과 d^2가 $d^1 < d^2$을 만족한다.

[정의-3] 다음의 조건을 만족하는 큐브 $C = c_{n-1} \cdots c_1 c_0$ $(c_j \in \{0, 1, *\},\ n-1 \geq j \geq 0)$을 LO **큐브**라고 정의한다.

- $c_j = *$ 이면, $j > k \geq 0$인 모든 k에 대해 $c_k = *$ 이다.

예를 들어, 0***는 LO 큐브이고, 0*0*는 LO 큐브가 아니다. 그리고, $S = \{C_1, C_2, \cdots, C_m\}$를 임의의 멀티캐스트 목적지를 나타내는 LO 큐브로 구성되는 순서 집합이라고 할 때, LO 큐브의 특성상 $C_j <_D C_k$ $(1 \leq j < k \leq m)$를 만족한다. 이러한 집합 S를 LO **큐브 집합**이라고 정의한다. 멀티캐스트 목적지가 {000, 010, 011, 100}인 경우의 예를 살펴본다. 목적지들은 LO 큐브 집합 {000, 01*, 100}으로 표시할 수 있다. 즉, 000, 01*, 100이 LO 큐브이고, $000 <_D 01* <_D 100$이다. 그러나, 목적지들을 표시하는 다른 큐브 집합 {*00, 01*}은 LO 큐브 집합이 아니다. *00가 LO 큐브가 아니며, 큐브간에 관계 $<_D$가 없기 때문이다. 본 논문에서는 LO 큐브 집합을 대상으로 한다. 그리고, 정의-1의 $S_{msb}(a, b)$는 a와 b가 LO 큐브인 경우에도 동일하게 정의할 수 있다. 예를 들어, $S_{msb}(01**, 00**) = 1$ 이다.

2.4 고장 모델과 용어 정의

본 논문에서는 SE가 고장인 경우를 다루고 있으며, $(n-1)$단과 0 단에 있는 SE는 고장이 아님을 가정한다. 이것은 메시지의 입력 또는 출력이 MIN의 양쪽 끝단에 있는 고장난 SE와 연결된 경우에, 라우팅이 불가능하기 때문이다.

[정의-4] $M(0)$과 $M(1)$은 최상위비트가 각각 0과 1인 주소들의 집합으로, $L(0)$과 $L(1)$은 최하위비트

가 각각 0과 1인 주소들의 집합으로 정의한다.

예를 들면, 주소 0101은 $M(0)$에 속하며, $L(1)$에도 속한다. 당연히, $M(0) \cap M(1) = \phi$ 이고, $L(0) \cap L(1) = \phi$ 이다.

[정의-5] $i\,(n\text{-}2 \geq i \geq 1)$ 단에 있는 하나의 SE를 $\alpha_{n-1}\cdots\alpha_{i+1}\beta_{i-1}\cdots\beta_0$으로 표시하며, MIN상의 모든 SE를 다음과 같은 두 개의 집합 F_0와 F_1로 구분한다.

- $F_0 = \{f = \alpha_{n-1}\cdots\alpha_{i+1}\beta_{i-1}\cdots\beta_0 \mid \alpha_{n-1} \oplus \beta_0 = 0\}$
- $F_1 = \{f = \alpha_{n-1}\cdots\alpha_{i+1}\beta_{i-1}\cdots\beta_0 \mid \alpha_{n-1} \oplus \beta_0 = 1\}$

또한, 만일 모든 고장 SE들이 집합 F_0에 속한 경우를 F_0 **고장**, 모든 고장 SE들이 집합 F_1에 속한 경우를 F_1 **고장**이라고 정의한다.

예를 들면, 4단으로 구성된 MIN의 모든 단에서 SE 000, 010, 101, 111은 $\alpha_{n-1} \oplus \beta_0 = 0$ 이므로 F_0에 속하며, SE 001, 011, 100, 110은 $\alpha_{n-1} \oplus \beta_0 = 1$ 이므로 F_1에 속한다. 본 논문에서 제안하는 고장 허용 멀티캐스팅 알고리즘은 MIN에서 F_0 고장과 F_1 고장이 발생한 경우를 처리한다.

[정의-6] 멀티캐스트 목적지를 나타내는 정렬된 큐브의 집합 $D = \{C_1, C_2, \cdots, C_m\}$는 다음과 같은 두 개의 분할 집합(partitioned set) $D0$과 $D1$로 구분한다.

- $D0 = \{C_i \mid C_i \subseteq M(0), 1 \leq i \leq m\}$
- $D1 = \{C_j \mid C_j \subseteq M(1), 1 \leq j \leq m\}$

[정의-7] $S = \{s^1, \cdots, s^j\}$가 $s^1 < s^2 < \cdots < s^j$인 출발지 주소들의 집합이고, $D = \{C_1, C_2, \cdots, C_k\}$가 $C_1 <_D C_2 <_D \cdots <_D C_k$인 목적지 LO 큐브 집합이라고 할 때, $S \Rightarrow {}^l\!D\ (1 \leq l \leq \min(j,k))$는 출발지 $s^m\ (1 \leq m \leq l)$에서 목적지 큐브 C_m으로 m개의 메시지들이 동시에 라우팅되는 것을 표현한다.

만일, 출발지 주소 집합 S가 {1000, 1010, 1100, 1110}이고 목적지 LO 큐브 집합 D가 {0001, 001*, 0111}일 경우, $S \Rightarrow^3 D$는 출발지 1000에서 목적지 큐브 0001로, 출발지 1010에서 목적지 큐브 001*로, 출발지 1100에서 목적지 큐브 0111로 각각 전송되는 3개의 메시지들이 동시에 라우팅됨을 의미한다.

3. 고장 허용 멀티캐스팅 알고리즘

본 장에서는 F_0 고장 또는 F_1 고장인 MIN에서의 멀티캐스팅 알고리즘을 기술한다. 임의의 멀티캐스트 목적지들은 하나의 LO 큐브 집합으로 표현이 가능하며, 알고리즘에서는 목적지 LO 큐브 집합을 사용한다. 이 알고리즘의 멀티캐스팅 예를 보이고, 알고리즘의 정확성을 증명한다.

3.1 멀티캐스팅 알고리즘

고장 허용 멀티캐스팅 알고리즘(FTM)은 (그림 2)와 같이 두 단계로 구성된다. 단계 1에서 목적지 큐브의 개수에 대응되는 크기($=2 \times k$)를 갖는 중간 출력 큐브 IC로 메시지를 복사하고, 단계 2에서 IC에 속한 노드들은 복사된 메시지를 실제 목적지 큐브들로 각각 라우팅을 한다.

가정

1. 모든 고장은 F_0 고장 또는 F_1 고장이다.
2. 멀티캐스트 목적지는 LO 큐브 집합으로 구성된다.
3. k는 $\max(2^{\lceil \log_2 |D0| \rceil}, 2^{\lceil \log_2 |D1| \rceil})$이다.

알고리즘

단계 1 : 출발지 $s = s_{n-1}\cdots s_1 s_0$에서 하나의 중간 출력 큐브 IC로 ($|IC| = 2k$) 메시지를 복사한다. 이 때, IC는 다음의 조건을 만족하는 임의의 큐브이다.

- F0 고장의 경우, $IC \subseteq M(\overline{s_0})$
- F1 고장의 경우, $IC \subseteq M(s_0)$

단계 2 : IC의 노드는 중간 출발지로서, 복사된 메시지를 대응되는 목적지 큐브로 다음과 같이 라우팅한다. 이 때, IC를 $S_0 \subseteq L(0)$과 $S_1 \subseteq L(1)$로 구분한다.

- F_0 고장의 경우, $S_0 \Rightarrow^{|D1|} D1$ 하고 $S_1 \Rightarrow^{|D0|} D0$ 함
- F_1 고장의 경우, $S_0 \Rightarrow^{|D0|} D0$ 하고 $S_1 \Rightarrow^{|D1|} D1$ 함

IC에서 $(2k - |D0| - |D1|)$ 개의 나머지 노드는 자신의 메시지를 무시한다.

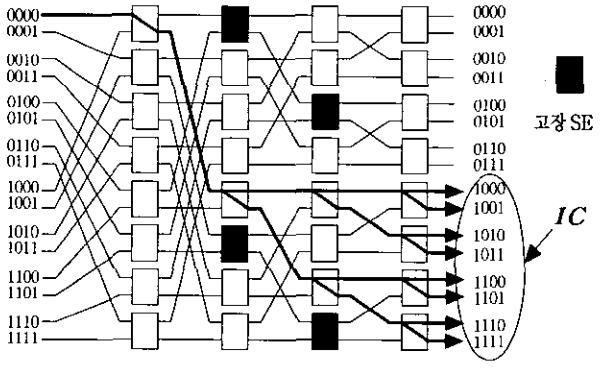
(그림 2) 고장 허용 멀티캐스팅 알고리즘(FTM)

3.2 알고리즘의 예

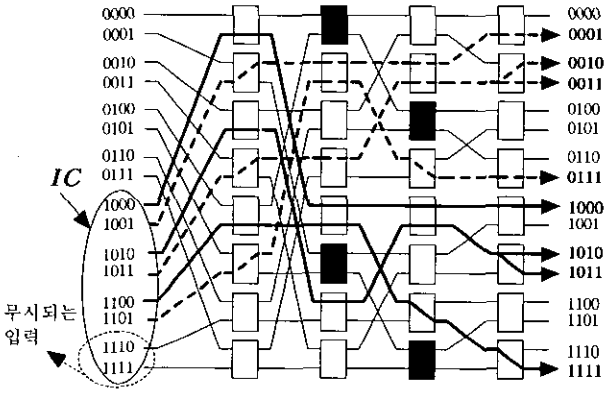
(그림 3)은 출발지 0에서 8개의 목적지 1, 2, 3, 7, 8, 10, 11, 15로 멀티캐스트 메시지를 전송할 때, FTM의 라우팅 예를 보이고 있다. 여기서 MIN은 2단에서 000, 101과 1단에서 010, 111 등 4개의 고장 SE를 갖는다. 즉, 모든 고장 SE는 F_0에 속해 있고, F_0 고장이다. 멀티캐스트 목적지는 정렬된 큐브 집합 {0001, 001*, 0111, 1000, 101*, 1111}로 표시되며, $D0$ = {0001, 001*, 0111}과 $D1$ = {1000, 101*, 1111}으로 구분된다. $|D0| = 3$ 이고 $|D1| = 3$이므로 k는 4이다. 단계 1은 (그림

3) (a)와 같이, 출발지 0에서 크기가 8(=2 × k)인 중간 출력 큐브 IC=1***로 메시지를 복사한다. 여기서, 출발지가 0(=0000), 즉 s_0이 0이므로, $IC \subseteq M(1)$ 인 IC를 선택한다.

(그림 3) (b)는 단계 2에서의 라우팅 경로를 보여준다. IC는 S_0={1000, 1010, 1100, 1110}과 S_1={1001, 1011, 1101, 1111}로 구분된다. 중간 출발지인 1000, 1010, 1100은 복사된 메시지를 각각 1000, 101*, 1111로 전송한다. 마찬가지로, 중간 출발지인 1001, 1011, 1101는 복사된 메시지를 각각 0001, 001*, 0111로 전송한다. S_0의 1110과 S_1의 1111은 자신의 메시지를 버린다. (그림 3)에서 보는 바와 같이, 모든 메시지는 고장 SE들을 통과하지 않으며, 더욱이 2단의 SE 010, 111과 1단의 SE 000, 101 등도 지나지 않음을 알 수 있다.



(a) 단계 1 (메시지의 복사)



(b) 단계 2 (목적지 큐브로 라우팅)

(그림 3) FTM의 두 단계 멀티캐스팅의 예

3.3 알고리즘의 정확성 증명

고장이 있는 MIN에서 FTM이 큐브 부호화 방식으로 표현된 임의의 멀티캐스트 메시지를 라우팅 할 수 있음을 증명하고자 한다. 먼저, FTM의 두 단계에서 모든 메시지가 고장 SE를 통과하지 않아야 한다. 그리고, 단계 1에서 목적지를 나타내는 LO 큐브의 수만큼 메시지를 복사가 가능해야 하며, 단계 2의 메시지간에 서로 충돌이 없어야 한다.

[보조정리 1] 메시지의 출발지를 $s = s_{n-1} \cdots s_1 s_0$, 목적지 큐브를 $d = d_{n-1} \cdots d_1 d_0$로 표시할 때, MIN에서 F_0 고장이면, $s_0 \neq d_{n-1}$인 메시지는 고장 SE를 통과하지 않으며, F_1 고장이면, $s_0 = d_{n-1}$인 메시지는 고장 SE를 통과하지 않는다.

증명 : 먼저, F_0 고장인 경우를 살펴본다. MIN의 특성-1에 따라, $i\,(n-2 \geq i \geq 1)$ 단에서 사용되는 SE는, $s_0=0$이고 $d_{n-1}=1$인 메시지이면 $1\gamma0$ ($|\gamma|=n-3$) 형태이며, $s_0=1$이고 $d_{n-1}=0$인 메시지이면 $0\gamma1$ ($|\gamma|=n-3$) 형태이다. 따라서, $s_0 \neq d_{n-1}$인 메시지는 F_0에 속한 고장 SE를 통과하지 않는다. F_1 고장인 경우도 같은 방식으로 증명할 수 있다. □

단의 수가 4인 MIN에서, 출발지가 $L(0)$에 속하고 목적지 큐브가 $M(1)$에 속한 메시지가 2단과 1단에서 통과하는 SE는 100 또는 110이며, 모두 F_1에 속해 있다. 따라서 이러한 메시지는 F_0에 속한 고장 SE를 통과하지 않는다. 마찬가지로 출발지가 $L(1)$에 속하고 목적지 큐브가 $M(0)$에 속한 메시지도 F_0에 속한 고장 SE를 통과하지 않는다.

[보조정리 2] 출발지가 각각 $L(0)$와 $L(1)$에 속한 임의의 두 메시지의 목적지가 서로 다르면, 두 메시지는 서로 충돌하지 않는다.

증명 : 두 메시지의 목적지를 각각 $a_{n-1} \cdots a_1 a_0$와 $b_{n-1} \cdots b_1 b_0$라고 하자. MIN의 특성-1에 따라, 두 메시지는 $j\,(n-1 \geq j \geq 1)$ 단에서 같은 SE를 통과하지 않는다. 만일, $a_j = b_j\,(n-1 \geq j \geq 1)$이면, 두 메시지는 0 단에서 같은 SE $a_{n-1} \cdots a_j$에서 만나지만, a_0과 b_0는 같을 수가 없으므로 그 SE에서 다른 출력 포트 a_0과 b_0로 라우팅된다. 따라서, 충돌이 발생되지 않는다. □

[보조정리 3] MC_k를 k-비트로 구성할 수 있는 큐브의 전체 개수라고 할 때, $MC_k \leq 2^{k-1}$이다.

증명 : k에 대한 수학적 귀납법으로 증명한다. $k=1$이면, $MC_k=1$이고 보조정리가 참이다. 보조정리가 k-1일 때 참이라고 가정하자. k-비트로 구성되는 큐

브를 $C_k^0 = \{d \mid d \in M(0)\}$와 $C_k^1 = \{d \mid d \in M(1)\}$로 구분하자. 그러면, 가정에 의해 $|C_k^0| \le 2^{k-2}$이고 $|C_k^1| \le 2^{k-2}$이다. 만일 두 큐브 $C_a = 0a_1a_2 \cdots a_{k-1} \in C_k^0$, $C_b = 1b_1b_2 \cdots b_{k-1} \in C_k^1$ ($a_j, b_j \in \{0, 1, *\}$, $1 \le j \le k-1$)이 있고, 모든 j ($1 \le j \le k-1$)에 대해 $a_j = b_j$ 이면, 그러한 두 큐브는 하나의 큐브 $*a_1a_2 \cdots a_{k-1}$로 병합된다. 따라서, $MC_k \le |C_k^0| + |C_k^1| \le 2^{k-1}$이다. □

[보조정리 4] a와 b가 $a < b$를 만족하는 n-비트 수일 때, $S_{lsb}(a, b) \le log_2(b-a)$ 이다.

증명 : $k = S_{lsb}(a, b)$라고 하자. 그러면, a와 b는 각각 $a_{n-1} \cdots a_k c_{k-1} \cdots c_0$ 와 $b_{n-1} \cdots b_k c_{k-1} \cdots c_0$ 로 표현할 수 있고, $b-a \ge 2^k$ 이다. 결과적으로, $S_{lsb}(a, b) = k \le log_2(b-a)$ 이다. □

[보조정리 5] $S = \{C_1, C_2, \cdots, C_m\}$을 정렬된 큐브 집합이라고 할 때, $1 \le a < b \le m$이면, $S_{msb}(C_a, C_b) \le n - log_2(b-a+1) - 1$이다.

증명 : $k = S_{msb}(C_a, C_b)$라고 하자. 그러면, $a < j < b$ 인 모든 j에 대해, $C_a <_D C_j <_D C_b$이므로, $S_{msb}(C_a, C_j) \ge k$이다. 따라서, 큐브 C_l ($a \le l \le b$)들의 k 개 최상위비트가 동일하다. 이러한 C_l의 수는 당연히 $b-a+1$ 이며, 또한 나머지 $n-k$ 비트를 고려할 때, $n-k$ 비트로 구성 가능한 큐브의 최대 개수 이하이다. 즉, 보조정리 3에 의해, $b-a+1 \le 2^{(n-k)-1}$ 이다. 따라서, $S_{msb}(C_a, C_b) \le n - log_2(b-a+1) - 1$ 이다. □

하나의 멀티캐스트 메시지를 전달할 경우, 알고리즘 FTM은 단계 1에서 한 개의 메시지를 전달하지만, 단계 2에서는 여러 개의 복사된 메시지가 서로 다른 입력단에서 원하는 목적지로 전달되어야 하기 때문에 이 메시지간에 충돌이 발생하지 않아야 한다. 정리 1은 여러 개로 복사된 메시지들 사이에도 충돌이 없음을 보여준다.

[정리 1] F_0 고장 또는 F_1 고장이 있는 MIN에서, 알고리즘 FTM은 LO 큐브 집합으로 표현된 임의의 멀티캐스트 메시지를 라우팅 할 수 있으며 메시지들간의 충돌 없이 전달한다.

증명 : F_1 고장은 F_0 고장과 동일하게 유추할 수 있으므로 F_0 고장에 대한 알고리즘의 정확성을 증명한다. 먼저, 임의의 멀티캐스트 목적지를 나타내는 LO 큐브 집합의 라우팅 가능성을 살펴본다. 단계 1을 보면, 보조정리 3에 의해 $|D0| \le 2^{n-2}$ 이고 $|D1| \le 2^{n-2}$ 이므로, $|IC| \le 2^{n-1}$ 이다. 또한 $|M(0)| = |M(1)| = 2^{n-1}$이다. 따라서 $|IC| \le |M(0)| = |M(1)|$ 이다. 단계 2를 보면, $|S_0| = |IC|/2 \ge |D0|$ 이고 $|S_1| = |IC|/2 \ge |D1|$ 이다. 그러므로, 다른 문제가 없다면 FTM은 임의의 목적지를 갖는 멀티캐스트 메시지를 라우팅 할 수 있다.

고장이 있는 MIN에서 멀티캐스팅시 발생 가능한 문제는 메시지가 고장 요소를 통과하거나 메시지간에 충돌이 일어나는 현상이다. 두 단계 모두 $L(0)$에 속한 출발지는 $M(1)$에 포함된 목적지로 라우팅하고, $L(1)$에 속한 출발지는 $M(0)$에 포함된 목적지로 라우팅한다. 그러므로, 보조정리 1에 의해 모든 메시지가 고장 SE를 통과하지 않는다. 그리고, 단계 1에서 하나의 메시지만 라우팅 되므로 당연히 메시지간의 충돌은 없다.

단계 2에서 임의의 두 메시지간의 충돌 가능성을 살펴보자. $S_0 \subseteq L(0)$, $S_1 \subseteq L(1)$이므로, 보조정리 2에 의해, $S_0 \Rightarrow^{|D1|} D1$의 메시지들과 $S_1 \Rightarrow^{|D0|} D0$ 의 메시지들 사이에는 충돌이 없다. $S_0 \Rightarrow^{|D1|} D1$ 에 해당되는 메시지들의 실질적 중간 출발지 집합을 $\{s^1, s^2, \cdots, s^{|D1|}\} \subseteq S_0$, 목적지 큐브 집합 $D1$을 $\{C_1, C_2, \cdots, C_{|D1|}\}$라고 하자. $1 \le j < k \le |D1|$ 이면, 보조정리 4에 의해, $S_{lsb}(s^j, s^k) \le log_2(s^k - s^j) = log_2(k-j) + 1$이다. 또한 보조정리 5에 의해, $S_{msb}(C_j, C_k) \le n - log_2(k-j+1) - 1$이다. 결과적으로, $S_{lsb}(s^j, s^k) + S_{msb}(C_j, C_k) < n$이다. 즉, 특성-2에 따라, $S_0 \Rightarrow^{|D1|} D1$ 의 메시지들 사이에 충돌이 없다. 또한, 같은 방식으로 $S_1 \Rightarrow^{|D0|} D0$ 의 메시지들 사이에 충돌이 없음을 증명할 수 있다. □

4. 결 론

본 논문에서는 대규모 다중 컴퓨터 구성에 사용되는 순환 구조를 갖는 MIN에서의 고장 허용 멀티캐스팅 알고리즘을 제안하였다. 제안된 알고리즘은 고장을 우회하여 멀티캐스팅을 수행하기 위해, MIN에서 메시지를 반복 순환시키는 기법을 사용하였다. 또한 큐브 부호화 방식을 사용하여 임의의 멀티캐스트 목적지를 표현하였고, 고장난

MIN을 두번 통과하여 멀티캐스트 메시지를 라우팅할 수 있다. 첫 번째 순환에서 임시 출력단으로 멀티캐스트 메시지를 복사한 후, 두 번째 순환에서 실제 목적지 큐브들로 라우팅한다. 또한, 스위칭 소자들을 두 개의 집합으로 구분하고, 한 집합에 속한 다수의 스위칭 소자들이 고장난 MIN에서 멀티캐스팅이 가능함을 증명하였다. 향후 연구 과제로는 본 알고리즘에서 허용할 수 있는 고장의 개수에 대한 분석과 알고리즘에 대한 성능 분석 및 평가 등이 있으며, 이에 대한 연구를 진행 중에 있다.

참 고 문 헌

[1] P. K. McKinley, Y. Tsai, and D. F. Robinson, "Collective Communication in Wormhole-Routed Massively Parallel Computers," IEEE Computer, Vol.28, No.12, pp.39-50, Dec. 1995.

[2] D. K. Panda, "Issues in Designing Efficient and Practical Algorithms for Collective Communication Wormhole- Routed Systems," Proc. of the Int'l Conf. on Parallel Processing, pp.8-15, Aug. 1995.

[3] C. B. Stunkel et al., "The SP2 Communication Subsystem," Technical report, IBM Thomas J. Watson Research Center, Aug. 1994.

[4] N. Koike, "NEC Cenju-3 : A Microprocessor-Based Parallel Computer," Proc. of the Int'l Parallel Processing Symposium, pp.396-401, Apr. 1994.

[5] G. B. Adams, D. P. Agrawal, and H. J. Siegel, "A Survey and Comparison of Fault-Tolerant Multistage Interconnection Networks," IEEE Computer, Vol.20, pp.14-27, Jun. 1987.

[6] S. J. Wang, "Distributed Routing in a Fault-Tolerant Multistage Interconnection Network," Information Processing Letters, Vol.63, No.4, pp.205-210, 1997.

[7] A. Varma and C. S. Raghavendra, "Fault-Tolerant Routing in Multistage Interconnection Networks," IEEE Transactions on Computers, Vol.38, No.3, pp.385-393, Mar. 1989.

[8] S. Chalasani, C. S. Raghavendra, and A. Varma, "Fault-Tolerant Routing in MIN-Based Supercomputers," Journal of Parallel and Distributed Computing, Vol.22, No.2, pp.154-167, Aug. 1994.

[9] N. Das and J. Dattagupta, "Two-Pass Rearrangeability in Faulty Benes Networks," Journal of Parallel and Distributed Computing, Vol.35, pp.191-198, Jun. 1996.

[10] R. Cusani and F. Sestini, "A Recursive Multistage Structure for Multicast ATM Switching," Proc. of IEEE Infocom, pp.1289-1295, Apr. 1991.

[11] X. Chen and V. Kumar, "Multicast Routing in Self-Routing Multistage Networks," Proc. of IEEE Infocom, pp.306-314, Apr. 1994.

[12] J. Park and H. Yoon, "Design and Performance Analysis of Multistage Interconnection Networks using a Recursive Multicast Algorithm," Int'l Journal of High Speed Computing, Vol.8, No.4, pp.347-362, 1996.

[13] C. S. Raghavendra, X. Chen, and V. P. Kumar, "A Two Phase Multicast Routing Algorithm in Self- Routing Multistage Networks," Proc. of Int'l Conference on Communications, pp.1612-1618, Jun. 1995.

[14] J. Park and H. Yoon, "Cost-Effective Algorithms for Multicast Communication in ATM Switches based on Self-Routing Multistage Networks," Computer Communications," Vol.21, pp.54-64, Feb. 1998.

[15] C. L. Wu and T.-Y. Feng, "On a Class of Multistage Interconnection Networks," IEEE Transactions on Computers, Vol.29. No.8, pp.694-702, Aug. 1980.

[16] N. Linial and M. Tarsi, "Interpolation between Bases and the Shuffle Exchange Network," European Journal of Combinatorics, Vol.10, pp.29-39, 1989.

[17] V. Chandramouli and C. S. Raghavendra, "Nonblocking Properties of Interconnection Switching Networks," IEEE Transactions on Communications," Vol.43, pp.1793-1799, 1995.

김 진 수

e-mail : jinsoo@kku.ac.kr

1979년~1983년 서울대학교 컴퓨터공학과 (학사)

1983년~1985년 KAIST 전산학과(석사)

1993년~1998년 KAIST 전산학과(박사)

1985년~2000년 한국전기통신공사 선임연구원

2000년~현재 건국대학교 컴퓨터응용학부 조교수

관심분야 : Interconnection Network, Parallel Computer Systems, High Speed Network, Network Security, Mobile Computing

박 재 형

e-mail : jhpark@etri.re.kr

1987년~1991년 연세대학교 전산학과 (학사)

1991년~1993년 KAIST 전산학과 (석사)

1993년~1997년 KAIST 전산학과 (박사)

1997년~1998년 KAIST 인공지능연구센터 Post-Doc 연수연구원

1998년~현재 ETRI 네트워크기술연구소 선임연구원

관심분야 : Parallel Processing, Interconnection Network, Multicast, ATM Switch, MPLS, TCP/IP, Wireless Protocol

김 명 균

e-mail : mkkim@uou.ulsan.ac.kr

1980년~1984년 서울대학교 컴퓨터공학과 (학사)

1984년~1986년 KAIST 전산학과(석사)

1990년~1996년 KAIST 전산학과(박사)

1986년~1989년 현대중공업 전산실

1989년~1998년 우석대학교 컴퓨터공학과

1998년~현재 울산대학교 컴퓨터정보통신공학부 조교수

관심분야 : Interconnection Network, Network Management. Parallel Processing