행렬 하이퍼큐브 그래프 : 병렬 컴퓨터를 위한 새로운 상호 연결망

최선아*, 이형옥, 임형석
전남대학교 전산학과

Matrix Hypercube Graphs : A New Interconnection Network for Parallel Computer

Sun-A Choi*, Hyeong-Ok Lee, Hyeong-Seok Lim
Dept. of Computer Science, Chonnam Univ.

Abstract - In this paper, we propose a matrix hypercube graph as a new topology for parallel computer and analyze its characteristics of the network parameters, such as degree, routing and diameter. N-dimensional matrix hypercube graph $MH(2,n)$ contains 2^{2n} vertices and has relatively lower degree and smaller diameter than well-known hypercube graph. The matrix hypercube graph $MH(2,n)$ and the hypercube graph Q_{2n} have the same number of vertices. In terms of the network cost, defined as the product of the degree and diameter, the former has n^2 while the latter has $4n^2$. In other words, it means that matrix hypercube graph $MH(2,n)$ is better than hypercube graph Q_{2n} with respect to the network cost.

1. 서론

최근 컴퓨터 제조 기술의 눈부신 발전과 보다 높은 성능을 요구하는 응용 분야의 폭발적인 증대로 인하여 일어나는 문제를 해결하기 위한 고성능 컴퓨터의 필요성이 절실히 요구되고 있다. 고성능 컴퓨터를 얻기 위한 가장 일반적이고 비용이 적게 드는 방법은 다중 컴퓨터를 설계하는 방법이다. 다중 컴퓨터의 각 노드는 개별 메모리, 메시지를 라우팅 할 수 있는 통신제어기와 다른 노드와의 통신을 위한 통신링크로 구성된다. 다중 컴퓨터의 위상(topology)은 전체 시스템의 성능을 결정하는데 중요한 역할을 한다[1,6].

상호 연결망은 각 프로세서들을 노드로, 프로세서들 사이에 통신 채널을 에지로 나타내는 무방향 그래프(undirected graph)로써 표현 될 수 있다. 상호 연결망을 평가하는 망척도로는 분지수(degree), 연결도(connectivity), 확장성(scalibility), 지름(diameter), 고장 허용도(fault tolerance) 및 대칭성(symmetric)등이 있다[2,5,8].

망척도 중 하드웨어의 비용과 관련된 분지수(degree)와 메시지의 전송시간과 관련된 지름(diameter)은 상호간에 상관관계를 갖고 있다[3,8].

연결망 G에 속한 임의의 노드 v의 분지수란 노드 v에 인접한 에지의 개수이다. G의 분지수는 $V(G)$에 속한 노드들의 분지수 중 최대값을 말한다. 그리고 지름이란 연결망 G에 속한 임의의 두 노드간의 최단 경로들 중 최대값이다. 일반적으로 연결망의 분지수를 확장하면 지름을 줄이게 되어 그 연결망에서의 처리량(throughput)을 높일 수 있는 장점이 있지만 병렬컴퓨터를 설계할 때 처리기의 핀 수가 늘어나게 되어 하드웨어의 비용이 증가하고 라우팅 제어가 복잡하게 되는 단점이 있다. 분지수가 작은 연결망은 하드웨어 비용은 줄어드는 반면 메시지 전송시간이 늘어나게 되어 연결망의 지연시간(latency time)이나 처리량이 나빠지는 단점이 있다. 이러한 특성 때문에 연결망의 비용[3,8]은 분지수 × 지름으로 평가하는데, 연결망이 같은 개수의 노드를 가질 때 분지수 × 지름 값이 작은 연결망이 좋은 특성을 갖는 것으로 분석한다.

이러한 상호 연결망으로는 트리, 메쉬[5], 하이퍼큐브[2,4], 스타 그래프[1] 등이 제안 되었다. 이들 중 가장 널리 알려지고 상용화된 것 중 하나가 하이퍼큐브이다. 하이퍼큐브는 각 노드의 주소를 n 비트 이진수로 표현했을 때 두 노드의 주소가 정확히 1 비트만 다른 노드 사이에 에지가 있다. 2^n개의 노드를 갖는 하이퍼큐브의 분지수와 지름은 각각 n이다. 하이퍼큐브의

특성은 연결망이 대칭이고 단순한 재귀적 구조를 가지고 있으며 링, 트리, 피라미드, 메쉬 등과 같은 다른 연결망 구조들이 효율적으로 임베딩될 수 있다는 장점이 있지만 노드수에 비해 지름이 크고, 연결망의 크기가 증가할수록 분지수가 늘어나는 단점이 있다[7,8]. 아래의 표에서 하이퍼큐브와 그 변형된 연결망들이 같은 수의 노드를 가질 때 분지수, 지름 및 연결망의 비용을 비교하였다.

표 1. 변형된 하이퍼큐브의 연결망 비용 비교

척도 / 연결망	노드수	분지수	지름	연결망 비용
Hypercube	2^{2n}	$2n$	$2n$	$4n^2$
Multiply-twisted-cube	2^{2n}	$2n$	$\lceil \frac{2n+1}{2} \rceil$	$2n^2$
$HCN(n,n)$	2^{2n}	$n+1$	$n+\lfloor \frac{n}{2} \rfloor+1$	$\frac{3n^2}{2}$
MCC	2^{2n+1}	$n+1$	$2n+2$	$2n^2$
$MH(2,n)$	2^{2n}	$n+2$	$n+1$	n^2

본 논문에서는 하이퍼큐브 보다 많은 수의 노드를 가지면서 상호 연결망의 비용[3,8] 즉, 분지수 × 지름 값이 하이퍼큐브 보다 작은 값을 갖는 새로운 상호 연결망으로 행렬 하이퍼큐브 그래프를 제안하고, 제안된 그래프에서의 라우팅 알고리즘과 지름을 분석하였다.

본 논문의 구성은 다음과 같다. 2장에서는 행렬 하이퍼큐브 그래프를 정의하고, 3장에서는 라우팅(routing) 알고리즘과 지름(diameter)을 분석하고, 마지막으로 결론을 맺는다.

2. 행렬 하이퍼큐브 그래프의 정의

행렬 하이퍼큐브 그래프 $MH(2,n)$는 $2n$개의 이진수로 구성된 2행 n열의 행렬 $\begin{bmatrix} s_1 & s_2 & \cdots & s_i & \cdots & s_n \\ s_{n+1} & s_{n+2} & \cdots & s_{n+i} & \cdots & s_{2n} \end{bmatrix}$ 형태로 노드를 표현하고 노드를 연결하는 에지는 다음과 같은 행렬로 표현된 노드 사이에 연결관계를 갖는다.

(1) 첫 번째 행에서 한 비트가 보수인 행렬

$$\begin{bmatrix} s_1 & s_2 & \cdots & \overline{s_i} & \cdots & s_n \\ s_{n+1} & s_{n+2} & \cdots & s_{n+i} & \cdots & s_{2n} \end{bmatrix} (\ 1 \le i \le n\ )$$

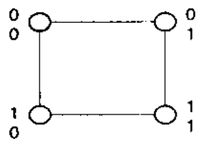
(2) 첫 번째 행의 보수와 두 번째 행이 교환된 행렬

$$\begin{bmatrix} s_{n+1} & s_{n+2} & \cdots & s_{n+i} & \cdots & s_{2n} \\ \overline{s_1} & \overline{s_2} & \cdots & \overline{s_i} & \cdots & \overline{s_n} \end{bmatrix}$$

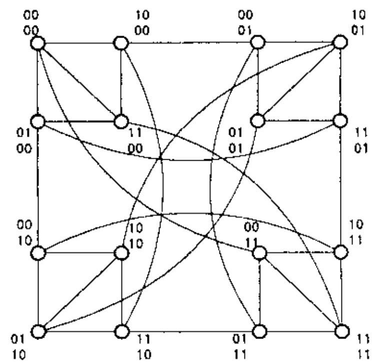
(3) 두 번째 행의 보수와 첫 번째 행이 교환된 행렬

$$\begin{bmatrix} \overline{s_{n+1}} & \overline{s_{n+2}} & \cdots & \overline{s_{n+i}} & \cdots & \overline{s_{2n}} \\ s_1 & s_2 & \cdots & s_i & \cdots & s_n \end{bmatrix}$$

위의 정의에 의해 $MH(2,n)$ 그래프는 $2n$개의 이진수로 표현할 수 있는 개수만큼 노드를 생성할 수 있으므로 노드의 개수는 2^{2n}개이고, 에지의 개수는 위의 3가지 조건에 의해 $n+2$개를 갖는 정규 연결망이다. 단, $n=1$인 $MH(2,1)$ 그래프는 에지 정의 조건에서 (1)조건에 만족하는 행렬과 (2)와 (3)의 조건에 의해 정의되는 행렬 중 하나가 동일하므로 분지수가 2이고 노드수가 4인 링 형태의 그래프이다. 위의 정의에 따라 구성된 $MH(2,1)$와 $MH(2,2)$ 그래프는 그림1과 같다.



(a) $MH(2,1)$ 그래프



(b) $MH(2,2)$ 그래프

그림 1. $MH(2,1)$와 $MH(2,2)$ 그래프

3. 라우팅 알고리즘과 지름

본 장에서는 행렬 하이퍼큐브 그래프 $MH(2,n)$에서의 라우팅 알고리즘을 제안하고 라우팅 알고리즘에 의한 지름(diameter)을 분석한다.

3.1 라우팅 알고리즘

행렬 하이퍼큐브 그래프 $MH(2,n)$의 노드 s와 d에서 노드 s를 원시 노드(source node)라 하고 노드 d를 목적 노드(destination node)라 할 때 노드 s와 d의 행렬을 아래와 같이 나타낸다. $MH(2,n)$ 그래프의 메시지 전송을 위한 라우팅은 행렬 연산을 통하여 원시 노드

s의 행렬을 목적 노드 d의 행렬로 변환하는 과정으로 나타낼 수 있다.

$$s=\begin{bmatrix} s_1 & s_2 & \cdots & s_i & \cdots & s_n \\ s_{n+1} & s_{n+2} & \cdots & s_{n+i} & \cdots & s_{2n} \end{bmatrix},$$

$$d=\begin{bmatrix} d_1 & d_2 & \cdots & d_i & \cdots & d_n \\ d_{n+1} & d_{n+2} & \cdots & d_{n+i} & \cdots & s_{2n} \end{bmatrix}$$

노드 s와 d를 exclusive-OR 연산을 한 결과를 r 행렬이라 하자. r 행렬에서 r_i = 0은 노드 s와 d의 비트 스트링에서 i번째 위치의 심볼이 서로 같음을 의미하고, r_i = 1은 서로 보수임을 의미한다.

$MH(2,n)$ 그래프에서 라우팅 개요는 다음과 같다.

노드 s 와 d 행렬의 각 행에 대한 8가지 경우의 exclusive-OR 연산값을 구한다. 8가지의 조건 중에서 1의 개수가 가장 적은 것을 선택하여 해당 경우를 수행한다. 만약 가장 작은 값을 가지는 경우가 둘 이상일 경우 다음의 우선 순위를 따른다. s_1과 s_2가 동시에 있을 때 s_1을 선택하고, d_1과 d_2가 동시에 있을 때 s_i을 기준으로 하여 i와 다른 행을 선택한다. 선택되어진 경우에 따라 수행되어지는 과정은 노드 s행렬의 1행의 원소가 d행렬의 1행의 원소와 동일하게 하려면 두 행간의 exclusive-OR 연산한 결과의 1의 개수만큼 원소를 교환한다. 노드 s의 1행과 노드 d의 2행이 동일한 원소이면 노드 s의 1행과 2행을 교환한 후에 노드 s의 1행에 노드 d의 1행과 동일하게 될 때까지 위의 과정을 반복한다. 노드 s의 한 행이 노드 d의 한 행과 동일하지 않으면 노드 s의 행에서 노드 d와 동일하지 않은 행을 1행에 위치하도록 하여 노드 d의 한 행과 동일한 행이 될 때까지 수행한다

위의 방법에 의한 라우팅 알고리즘과 알고리즘에서 사용하는 수식은 다음과 같다.

s_i : s 행렬의 i행의 원소.

$\overline{s_i}$: s 행렬의 i행의 원소에 대한 보수.

$s_i \rightarrow d_i$: s 행렬 i행을 d 행렬 i행의 원소로 변환.

$s_1 \leftrightarrow s_2$: s 행렬 1행과 s 행렬 2행 교환.

Operator ⇒ case 1 : $s_1 \oplus d_1$, case 2 : $s_1 \oplus d_2$,
case 3 : $s_2 \oplus d_1$, case 4 : $s_2 \oplus d_2$,
case 5 : $\overline{s_1} \oplus d_1$, case 6 : $\overline{s_1} \oplus d_2$
case 7 : $\overline{s_2} \oplus d_1$, case 8 : $\overline{s_2} \oplus d_2$,

Route(s, d)

Operator = $s_i \oplus d_i$ (i=1, 2)

$$k=\min\left| \begin{matrix} s_1 \oplus d_1, & s_1 \oplus d_2, & s_2 \oplus d_1, & s_2 \oplus d_2 \\ \overline{s_1} \oplus d_1, & \overline{s_1} \oplus d_2, & \overline{s_2} \oplus d_1, & \overline{s_2} \oplus d_2 \end{matrix} \right|$$

/* 8가지 경우를 check한 최소값 */

if(k값을 갖는 Operator가 둘 이상일 때)

$s_1 \oplus d_i$ 선택 : 동시에 s_1과 s_2가 Operator에 포함된 경우.

d_1 또는 d_2 선택 : 동시에 d_1과 d_2가 있을 때 s_i의 i와 다른 행을 선택.

case Operator of { /* 8가지 case 수행 */

1 : $s_1 \rightarrow d_1;\ s_1 \leftrightarrow \overline{s_2};$
$s_1 \rightarrow \overline{d_2};\ \overline{s_1} \leftrightarrow s_2;$

2 : 7 : $s_1 \rightarrow d_2;\ s_1 \leftrightarrow \overline{s_2};$
$s_1 \rightarrow d_1;$

3 : 6 : $s_1 \rightarrow \overline{d_2}\ ;\ \overline{s_1} \leftrightarrow s_2;$
$s_1 \rightarrow d_1;$

4 : if ($s_2 \oplus d_2$=0) then $s_1 \rightarrow d_1;$
else { $s_1 \rightarrow \overline{d_1};\ \overline{s_1} \leftrightarrow s_2;$
$s_1 \rightarrow d_2;\ s_1 \leftrightarrow \overline{s_2};$ };

5 : $s_1 \rightarrow \overline{d_1};\ s_1 \leftrightarrow \overline{s_2};$
$s_1 \rightarrow d_2;\ s_1 \leftrightarrow \overline{s_2};$

8 : $s_1 \leftrightarrow \overline{s_2};\ s_1 \rightarrow d_2;$
$s_1 \leftrightarrow \overline{s_2};\ s_1 \rightarrow d_1;$

}

예를 들어 $MH(2,8)$ 그래프의 노드 s와 노드 d의 행렬이 다음과 같을 때, 노드 s에서 d로 라우팅 알고리즘에 의한 수행 과정에서 사용되는 에지와 행렬 변환 과정은 다음과 같다.

$$s = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}\ d = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

$s_1 \oplus d_1$=2, $s_1 \oplus d_2$=4, $s_2 \oplus d_1$=1, $s_2 \oplus d_2$=1,

$\overline{s_1} \oplus d_1$=4, $\overline{s_1} \oplus d_2$=2, $\overline{s_2} \oplus d_1$=5, $\overline{s_2} \oplus d_2$=5

위의 8가지 경우를 연산하여 얻은 최소값은 1이고, 두 개의 경우에 해당되므로 우선 순위에 의해 $s_2 \oplus d_1$을 선택하게 되므로 case 3을 수행한다.

$$\begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \xrightarrow{s_1 \rightarrow \overline{d_2}} \begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \xrightarrow{s_1 \rightarrow \overline{d_2}}$$

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \xrightarrow{\overline{s_1} \leftrightarrow s_2} \begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix} \xrightarrow{s_1 \rightarrow d_1}$$

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \end{bmatrix}$$

3.2 지름

지름은 메시지가 어떤 노드에서 다른 노드까지 최단 경로로 전송될 때 지나는 통신 링크의 최대 개수를 나타낸다.

정리 행렬 하이퍼큐브 그래프 $MH(2,n)$의 지름은 $n+1$ 이다.

증명 원시노드 s의 행렬을 목적노드 d의 행렬로 변환하는 과정은 에지 연결 관계를 통하여 이루어지므로, 행렬 변환 과정에서 사용되는 에지의 최대 연산횟수를 통하여 분석한다. 노드 s의 1행에서 노드 d의 어떤 한 행과 동일한 행을 생성할 때 노드 s의 1행과 노드 d의 각 행과 보수인 원소의 개수가 k개라 하자. 노드 s의 1행에서는 한 비트씩 다른 행렬과 연결되므로 노드 d의 1행과 2행으로의 각 원소 변환 과정은 k번씩 수행된다. 노드 s의 1행에서의 연산 외에 추가적으로 발생하는 에지가 2가지가 있다. 첫 번째는 노드 s의 1행에서의 연산이 이루어지기 전에 노드 s의 1행과 2행을 교환하는 경우이고, 두 번째는 노드 s의 1행에서 노드 d의 1행이나 2행으로의 연산이 이루어진 후 s의 2행을 s의 1행의 위치로 교환하는 경우가 있다. 그러므로 에지 연산은 $2k+1$번 수행 가능하다. 이때 k값이 한 행을 구성하는 원소의 $n/2$일 때 연산 회수는 $2(n/2)+1 = n+1$이고, 노드 s와 노드 d의 대각선에 위치한 한 행의 원소가 모두 다를 때 즉, $k=n$일 때는 한 행에서만 n번 수행하므로 $k+1 = n+1$이 된다. □

상호 연결망을 평가하는 망척도 중 연결망의 비용 평가는 분지수 × 지름이 작은값을 가지면서 많은 수의 노드를 갖는 상호 연결망이 좋은 연결망이라 하겠다. 본 논문에서 제시하는 $MH(2,n)$ 그래프는 분지수 $n+2$와 지름 $n+1$을 가지며 연결망 비용 즉, 분지수 × 지름은 n^2+3n+2이다. 행렬 하이퍼큐브 그래프 $MH(2,n)$와 같은 노드 수를 갖는 하이퍼큐브 Q_{2n}는 분지수가 $2n$이고 지름도 $2n$인 연결망으로 분지수 × 지름은 $4n^2$ 이다. 그러므로 행렬 하이퍼큐브 그래프 $MH(2,n)$ 그래프가 연결망 비용 측면에서 하이퍼큐브 Q_{2n} 보다 우수한 결과를 가진다.

4. 결론

다중 컴퓨터 연결망은 전체 시스템의 성능 및 확장성에 큰 영향을 미치는 것으로 병렬 처리 컴퓨터 개발을 위해 그 중요성이 증가하고 있다. 다중 컴퓨터 연결망에서 연결망을 구성하는 처리기의 하드웨어 비용에 영향을 주는 분지수와 연결망의 메시지 전송 시간과 관련된 지름은 중요한 척도이다. 즉, 연결망의 분지수와 지름이 작은 값을 가지면서 많은 수의 노드를 갖는 연결망이 좋은 특성을 갖는 상호 연결망이라 할 수 있다. 지금까지 제안된 연결망에서 하이퍼큐브 그래프는 여러 가지 망 척도에서 우수한 특성을 갖는 연결망이다.

본 논문에서는 다중 컴퓨터 연결망으로 행렬 하이퍼큐브 그래프 $MH(2,n)$를 제안하였다. $MH(2,n)$ 그래프는 2^{2n}개의 노드를 갖고 분지수와 지름이 작은 값을 갖는 상호 연결망이다. $MH(2,n)$ 그래프는 분지수 $n+2$와 지름 $n+1$을 갖고, $MH(2,n)$ 그래프와 같은 노드 수를 갖는 하이퍼큐브 Q_{2n}는 분지수가 $2n$이고 지름도 $2n$인 연결망으로 분지수 × 지름에 대한 척도는 각각 n^2과 $4n^2$으로 행렬 하이퍼큐브 $MH(2,n)$가 하이퍼큐브 Q_{2n}보다 우수함을 의미한다. 향후의 연구 과제는 행렬 하이퍼큐브 연결망에서 방송 알고리즘을 제안하고 하이퍼큐브로의 임베딩에 대한 연구가 필요하다.

참고 문헌

[1] S. B. Akers and B. Krishnamurthy, "A Group-theoretic Model for Symmertric Interconnection Network," IEEE Trans. Comput., Vol. 38, No. 4, pp. 555-565, 1989.

[2] S. B. Akers, D. Harel, and B. Krishnamurthy, "The Star Graph: An Attractive Alternative to the n-Cube," Proc. Int. Conf. on Parallel Processing, pp. 393-400, Aug. 1987.

[3] L. N. Bhuyan and D. P. Agrawal, "Generalized Hypercube and Hyperbus Structures for a Computer Network," IEEE Trans. Comput., Vol. c-33, No. 4, pp. 323-333, Apr. 1984.

[4] F. Harary, J. P. Hayes, and H.-J. Wu, "A Survey of the Theory of Hypercube Graphs," Comput. Math. Appl., Vol. 15, pp. 277-289, 1988.

[5] M. S. Krishnamoorthy and B. Krishnamurthy, "Fault Diameter of Interconnection Networks," Comput. Math. Appl. Vol. 13, pp. 577-582, 1987.

[6] S. Lakshmivarahan, Jung-Sing Jwo, and S. K. Dhall, "Symmetry in Interconnection Networks based on Cayley Graphs of Permutation Groups: A Survey," Parallel Computing, Vol. 19, pp. 361-407, 1993.

[7] Y. Saad and M. H. Schutltz, "Topological Properties of Hypercubes," IEEE Trans. Comput., Vol. 37, pp. 867-872, 1988.

[8] John A. Stankovic, "A Perspective on Distributed Computer Systems," IEEE Trans. Comput., Vol. c-33, No. 12, pp. 1102-1115, December 1984.