

# 하이퍼-토러스 : 3차원 하이퍼큐브 기반의 새로운 토러스 네트워크

## (Hyper-Torus: A New Torus Network based on 3-dimensional Hypercube)

기우서<sup>†</sup>      김정섭<sup>\*\*</sup>      이형욱<sup>\*\*\*</sup>      오재철<sup>\*\*\*\*</sup>  
 (WooSeo Ki)      (JeongSeop Kim)      (HyungOk Lee)      (JaeChul Oh)

**요약** 본 논문에서는 3차원 하이퍼큐브 Q3를 기본모듈로 갖는 새로운 토러스 네트워크를 제안한다. 제안한 하이퍼-토러스는 분지수 4를 갖고, 확장성, 지름이 좋은 연결망이다. 토러스 부류를 망비용 관점에서 비교하면 하이퍼-토러스는  $1.4\sqrt{N}+16$ 으로 토러스의 망비용  $4\sqrt{N}$ 보다 대략 65% 개선되었고, 허니컴 토러스의 망비용  $2.45\sqrt{N}$ 보다 대략 50% 개선된 값이다. 이러한 결과는 하이퍼-토러스가 기존의 메시 부류보다 망비용 관점에서 우수함을 의미한다.

키워드 : 병렬 컴퓨터, 상호연결망, 하이퍼-토러스, 하이퍼큐브, 토러스

**Abstract** In this paper, we propose the new torus network which has the hypercube Q3 as the basic module. The proposed Hyper-torus has the degree 4, and is the network which has the scalability, and the fine diameter. If we compare the class of the torus in the viewpoint of network cost, the hyper-torus with  $1.4\sqrt{N}+16$  is proved to be approximately 65% than the torus with  $4\sqrt{N}$  and 50% than the honeycomb with  $2.45\sqrt{N}$ . This result means that hyper-torus is better for the class of the existing mesh in the viewpoint of network cost.

**Key words** : parallel computer, interconnection network, hyper-torus, hypercube, torus

### 1. 서론

상호연결망은 프로세서의 집합, 지역메모리 그리고 프로세서간의 데이터 전송을 위한 통신링크로 구성되어 있다. 상호연결망은 그래프  $G=(V,E)$ 로 모델링될 수 있다. 각각의 프로세서  $P_i$ 는 노드집합  $V$ 의 원소이고 두

개의 프로세서  $P_i$ 와  $P_j$ 는 통신링크( $P_i, P_j$ )에 의해 연결되어 있다. 상호연결망을 그래프로 모델링하면 프로세서는 노드(node)로 통신링크는 에지(edge)로 사상된다. 노드  $P_i$ 와 이웃한 노드의 수를 그 노드의 분지수(degree)라고 한다. 상호연결망에서 분지수는 하드웨어적인 비용을 의미한다.

상호연결망에서 지름(diameter)은 임의의 두 노드 사이의 최단경로 길이 중 최대값으로 정의되고, 연결망의 소프트웨어적인 처리 비용을 의미한다. 상호연결망의 지름과 분지수는 역 상관관계(trade-off)에 있다. 일반적으로 분지수를 증가시키면 지름은 짧아지고, 상대적으로 분지수를 적게하면 지름은 길어진다. 따라서 상호연결망의 망비용(network cost)은 분지수×지름 값으로 정의된다[1-4]. 상호연결망이 큰 지름 값을 갖는다는 것은 메시 전송 과정에서 교착 가능성이 커져 통신지연을 발생할 수 있으므로 상호연결망이 효율적이지 못하게 됨을 의미한다. 지금까지 다중컴퓨터를 위한 상호연결망의 망비용을 개선한 다양한 위상들이 발표되었다. 이러한 위상을 분지수 기준으로 분류하면 노드수가 증가해도 분지수가 고정된 메시(토러스 포함) 부류와 노드수가 증가함

---

† 정 회 원 : 순천대학교 컴퓨터학과  
k00165@lycos.co.kr  
 \*\* 학생회원 : 한국교원대학교 수학교육과  
is2subi@hanmail.net  
 \*\*\* 정 회 원 : 순천대학교 컴퓨터교육과 교수  
oklee@sccnu.ac.kr  
 \*\*\*\* 종신회원 : 순천대학교 컴퓨터학과 교수  
ojc@sccnu.ac.kr  
 논문접수 : 2008년 8월 18일  
 심사완료 : 2009년 1월 21일

Copyright©2009 한국정보과학회 : 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다. 이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.  
 정보과학회논문지 : 시스템 및 이론 제36권 제3호(2009.6)

에 따라 분지수가 증가하는 하이퍼큐브(hypercube) 부류와 스타 그래프(star graph) 부류로 분류할 수 있다.

메시 부류의 연결망들은 연속된 다각형 모양으로 다각형의 꼭짓점에 노드가 있고 변 끝에 있는 두 꼭짓점 사이에 에지가 있다. 사각형 모양의 메시[5], 메시에 랩어라운드(wraparound) 에지를 연결한 토러스(torus)[6], 메시의 구조에 에지를 사선으로 연결시킨 다이아고날 메시(diagonal mesh)[6], 육각형을 6개의 삼각형으로 분할한 헥사고날 메시(hexagonal mesh)[7], 육각형 모양의 허니컴 메시(honeycomb mesh)[8], 허니컴 메시에 랩어라운드 에지를 연결한 허니컴 토러스[8],  $(d,k)$ -그래프 문제를 이용한 피터슨-토러스[9] 등이 있다. 메시 부류의 연결망은 망의 확장이 쉽고, 낮은 차원의 메시는 상용화하기 쉬운 장점이 있다.

하이퍼큐브 부류에는 망의 노드수를 정확하게 두 배씩 확장하면서 확장된 노드와 기존의 노드를 각각 일대일로 연결하는 하이퍼큐브[10], 하이퍼큐브에서 지름 거리에 있는 노드들 사이에 에지를 추가한 폴디드(folded) 하이퍼큐브[11], 하이퍼큐브의 에지를 틀어서 만든 Twisted  $n$ -cube[1]와 multiply-twisted-cube[3], 그리고 그레이코드(gray code)를 이용하여 에지 연결을 정의한 그레이큐브[12] 등이 있다. 스타그래프 부류는 스타그래프[2], 매크로-스타(macro-star)[13], 전위(transposition) 그래프[14], 팬케익(pancake) 그래프[15] 등이 있다.

기존의 연결망을 이용하여 새로운 상호연결망을 만드는 방법에는 하이퍼큐브, 스타그래프, 피터슨그래프 등과 같이 기본이 되는 그래프를 재귀적으로 확장하는 계층적(hierarchical) 연결망과 두 개의 기본 그래프를 카티션곱(cartesian product)하여 만드는 카티션곱 연결망이 있다. 계층적 연결망에는  $n$ -HHC(hierarchical hypercube)[16], HFN(hierarchical folded-hypercube network)[17], HS(hierarchical star)[18], HFP $_{n,k}$ (hierarchical folded petersen)[19], RPSCH(Ringed Petersen Spheres Connected hypercube)[20] 등이 있다. 카티션곱 연결망에는 HPN(hyper-Petersen network), FPN(folded Petersen cube network), MCT(mesh-connected-trees), PSN(product-shuffle network), hyper-deBruijn network, star-cube network, hyperstar 등이 있다[21].

임의의 두 개의 그래프를  $G_1(V_1, E_1)$ ,  $G_2(V_2, E_2)$ 이라고 하고, 두 그래프를 카티션 곱한 그래프를  $G(V, E)$ 라고 하면,  $G=G_1 \times G_2$ 이다.  $G$ 의 노드  $V = \{ \langle x_1, x_2 \rangle \mid x_1 \in V_1, x_2 \in V_2 \}$ 이고,  $G$ 의 에지  $E = \{ \langle \langle x_1, y_1 \rangle, \langle x_2, y_2 \rangle \rangle \mid (x_1 = x_2 \text{ and } (y_1, y_2) \in E_1) \text{ or } (y_1 = y_2) \text{ and } (x_1, x_2) \in E_2 \}$ 이고,  $G_1$ 의 에지와  $G_2$ 의 에지를 그대로 갖기 때문에 그래프  $G$ 의 분지수는  $G_1$ 의 분지수 +  $G_2$ 의 분지수이다.

토러스는 메시에 랩어라운드 에지를 추가하여 노드

대칭이고 재귀적인 성질을 가지고 있다. 분지수가 4로 고정되어 있어 상용화가 쉽고 망의 확장이 쉬운 장점이 있어 Cray T3D, J-Machine(MIT), M-Machine(MIT), Symult 2010, Stanford Dash, IntelParagon, Intel Touchstone Delta 등 여러 가지 시스템으로 상용화되었다[22]. 하이퍼큐브의 노드 수는 2의 지수 승이고 분지수는  $\log_2 N$ ( $N$ 은 노드수)이다. 하이퍼큐브는 노드 및 에지 대칭이고 재귀적인 방법으로 확장이 가능하다. 병렬처리의 다양한 알고리즘에서  $2^d$ 개의 노드를 가지는  $d$ 차원 하이퍼큐브는 시간복잡도  $d$ 를 가지는 간단한 해결책을 가지고 있다[23]. 또한 선형배열, 링, 메시, 트리 등이 시뮬레이션될 수 있다. 이러한 장점 때문에 하이퍼큐브는 Intel iSPC, iSPC/2, NCUBE 부류, Cosmic Cube, Symult S-series, CM-1 그리고 CM-2 등으로 상용화되었다[17].

본 논문에서는 카티션곱 네트워크와 비슷해 보이지만 새로운 방법으로 하이퍼큐브와 토러스를 결합하는 하이퍼-토러스 네트워크를 설계하였다. 하이퍼-토러스의 설계 방법은 토러스의 노드에 3차원 하이퍼큐브를 대치하고, 3차원 하이퍼큐브의 노드에 에지를 1개씩 추가하여 네트워크를 구성한다. 카티션곱 네트워크와의 차이점은 분지수이다. 하이퍼-토러스의 분지수는 3차원 하이퍼큐브의 분지수보다 1 증가한 분지수 4를 갖는다. 하이퍼-토러스는 토러스의 망 비용을 개선하였고, 확장성, 짧은 지름, 적은 망비용 등의 우수한 성질을 갖는다. 또한, 기본모듈로 3차원 하이퍼큐브를 이용하므로 하이퍼큐브에서 개발된 많은 알고리즘을 토러스 구조에서 활용할 수 있을 것으로 기대한다. 본 논문의 구성은 다음과 같다. 2장 관련연구에서는 토러스와 3차원 하이퍼큐브에 대해서 살펴보고, 3장에서는 하이퍼-토러스를 정의하고, 단순 라우팅 알고리즘, 최적 라우팅 알고리즘을 제안하고, 지름, 망 비용 그리고 이분 폭을 분석하였고 헤일톤 사이클을 제시하였다. 마지막으로 결론을 맺는다.

## 2. 관련연구

상호연결망은 노드수가 증가함에 따라 분지수가 일정한 함수관계를 가지고 증가하는 연결망과 노드수의 증가와 관계없이 분지수가 일정한 개수를 갖는 연결망으로 분류할 수 있다. 분지수가 일정한 연결망은 대표적으로 메시와 토러스가 있고, 분지수가 증가하는 연결망은 대표적으로 하이퍼큐브와 스타 그래프가 있다.

토러스 구조는 메시 구조에서 가장 자리에 있는 노드들에 랩어라운드 에지를 추가하여 동일한 분지수를 갖는 정규연결망 형태를 갖는 구조이다. 메시 부류의 연결망을 망비용 관점에서 비교하면 다음과 같다.  $n \times n$  메시는 분지수 4, 노드수  $N=n^2$ 이고 망 비용은  $8\sqrt{N}$ 이다. 토러

스는 메시의 행과 열 각각에서 링 형태를 이루도록 램어라운드 에지를 추가하여 구성한 연결망으로, 지름을 메시의 1/2로 줄였고, 망 비용도 1/2로 줄였다.  $n \times n$  토러스는 분지수 4, 노드수  $N=n^2$ 이고 망비용은  $4\sqrt{N}$ 이다[6].

헥사고널(hexagonal) 메시는 삼각형을 모자이크(tessellation) 형태로 연속하여 배치하면서 교차점을 노드로 한다.  $n$ 차원 헥사고널 메시는 분지수 6, 노드수  $N=3n^2-3n+1$ 이고 망비용은  $6.93\sqrt{N}$ 이다. 헥사고널 토러스는 헥사고널 메시에 램어라운드 에지를 추가 하였으며, 분지수 6, 노드수  $N=3n^2-3n+1$ 이고 망 비용은  $3.46\sqrt{N}$ 이다[7].

허니컴(honeycomb) 메시는 중심에 있는 육각형의 각 변에 육각형을 배치하면서 네트워크를 확장하는 구조로서 분지수 3, 노드수  $N=6n^2$ 이고, 망 비용은  $4.90\sqrt{N}$ 이다. 허니컴 토러스는 허니컴 메시에 램어라운드 에지를 추가하였고, 분지수 3, 노드수  $N=6n^2$ , 망비용은  $2.45\sqrt{N}$ 이다[8].

2D 토러스 부류를 확장한 3D 토러스 구조로 3D 헥사고널 토러스[22], 3D 허니컴 토러스 구조가 제안되었다. 3D 헥사고널 토러스와 3D 허니컴 토러스 구조는 기존의 2D 토러스 구조에서 각 노드에 2개 이상의 에지를 추가하여 네트워크를 구성하였다.

$n$ -차원 하이퍼큐브  $Q_n$ 는  $2^n$ 개의 노드와  $n2^{n-1}$ 개의 에지로 구성된다. 각 노드의 주소는  $n$ -비트 이진수로 표현될 수 있고, 임의의 두 노드 주소가 정확히 1 비트만 다를 때 그들 사이에 에지가 있으므로, 하이퍼큐브  $Q_n$ 는 분지수가  $n$ 인 정규 그래프이다. 하이퍼큐브  $Q_n$ 에서 위의 정의와 같이 주소를 붙이는 방법은  $n!2^n$ 이다[10]. 하이퍼큐브  $Q_n$ 는 다음과 같은 단순한 재귀적 구조를 가지고 있으므로 확장성이 매우 좋다. 하이퍼큐브  $Q_n$ 은 그래프의 곱을 이용하여 다음과 같이 재귀적으로 정의될 수 있다.

$$Q_1=K_2,$$

$$Q_n=K_2 \times Q_{n-1} \text{ (단, } K_2 \text{는 노드가 두 개인 완전 그래프)}$$

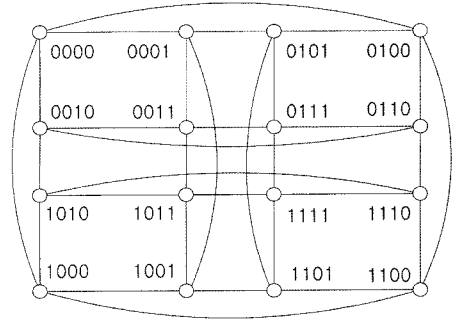


그림 2 4-차원 하이퍼큐브  $Q_4$

### 3. 하이퍼-토러스 네트워크 정의 및 성질

#### 3.1 하이퍼-토러스 연결망 정의

하이퍼-토러스  $QT(m,n)$ , ( $m,n \geq 2$ )는 3차원 하이퍼큐브를 기본모듈로 하고 기본모듈  $m$ ( $x$ 축)  $\times$   $n$ ( $y$ 축)개를 격자점에 배치하여 에지 정의에 의해 연결한다. 그림 3은 에지 정의에 의해 연결된 하이퍼-토러스  $QT(m,n)=(V_{qt}, E_{qt})$ 를 보여주고 있다. 본 논문에서는 하이퍼-토러스를 그림 4와 같이 2차원 그래프에서 사상시켜 설명한다. 3차원 하이퍼큐브를 기본모듈이라 하고, 기본모듈은  $x$ 축과  $y$ 축의 교차점에 배치한다. 기본모듈의 주소는  $(x,y)$ 로 표현하고, 노드 주소는  $(x,y,q)$ 로 나타낸다.  $x$ 는 기본모듈의  $x$ 축 좌표,  $y$ 는 기본모듈의  $y$ 축 좌표,  $q$ 는 기본모듈인 3차원 하이퍼큐브 그래프에 있는 노드의 주소이다. 하이퍼-토러스  $QT(m,n)$ 의 노드 정의는 다음과 같다.

$$V_{qt}=\{(x,y,q), 0 \leq x < m, 0 \leq y < n, q_i \in \{0,1\}, i=0,1,2\}$$

$QT(m,n)$ 의 에지는 내부 에지와 외부 에지로 나눌 수 있다. 내부 에지는 같은 기본모듈에 있는 노드를 연결하는 에지로서 3차원 하이퍼큐브 에지이다. 외부 에지는 서로 다른 기본모듈에 있는 노드를 연결하는 에지이다. 외부 에지의 정의는 다음과 같다. 심벌 '%'는 나머지 연산자이다.

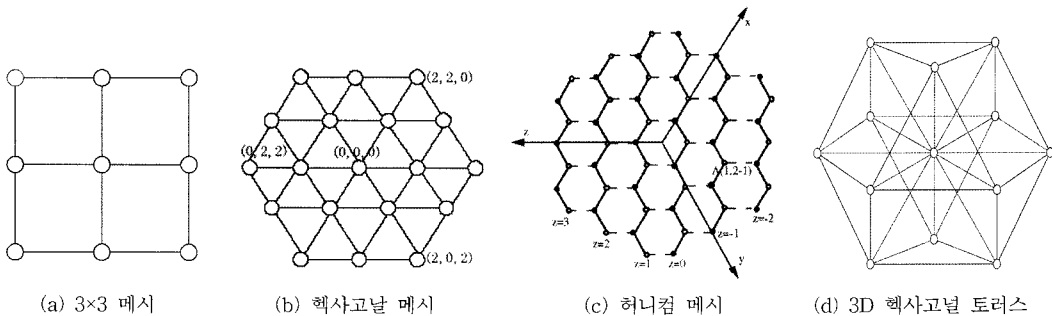


그림 1 토러스 네트워크

- ① 세로 예지:  $((x,y,101), (x,(y+1)\%n,001))$ 과  $((x,y,001), (x, (y-1+n)\%n, 101))$
- ② 가로 예지:  $((x,y,111), ((x+1)\%m,y,011))$ 과  $((x,y,011), ((x-1+m)\%m,y,111))$
- ③ 사선 예지:  $((x,y,110), ((x+1)\%m,(y+1)\%n,010))$ 과  $((x,y,010), ((x-1+m)\%m,(y-1+n)\%n, 110))$
- ④ 역사선 예지:  $((x,y,000), ((x-1)\%m,(y+1)\%n, 100))$ 과  $((x,y,100), ((x+1+m)\%m,(y-1)\%n,000))$

하이퍼-토러스의 정의에 의해 기본모듈 사이의 외부예지를 포함한 하이퍼-토러스  $QT(2,2)$ 는 그림 3과 같다.

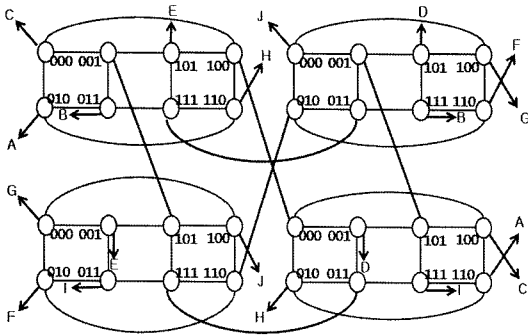


그림 3 하이퍼-토러스  $QT(2,2)$

그림 4는 하이퍼-토러스  $QT(5,5)$ 에서 기본모듈을 격자점으로 표현하였다. 기본모듈의 각 노드는 이웃하는 기본모듈의 노드들과 완전그래프 형태로 연결되어 있다. 가장 자리에 위치한 기본모듈들은 그림의 복잡성 때문에 몇몇 예지를 생략하였고, 나머지 기본모듈들은 모든 예지들이 표현되어 있다. 가장자리의 기본모듈들은 랩어라운드 예지가 생략되어 있으나 4개의 꼭짓점에 위치하는 기본모듈들은 몇 개의 랩어라운드 예지가 그려져 있다.

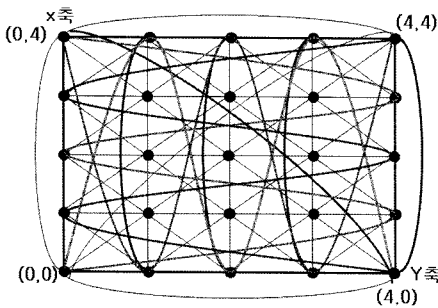


그림 4 하이퍼-토러스  $QT(5,5)$

하이퍼-토러스  $QT(m,n)$ 은 노드수  $8mn$ , 예지 수  $16mn$ , 그리고 분지수 4를 가지는 정규그래프(regular graph)이다. 임의의 노드  $U$ 로부터 임의의 노드  $V$ 까지 경로가

있다고 가정하면, 그 경로에 사용된 외부 예지의 수를 외부 경로길이라 하고, 내부 예지의 수는 내부 경로길이라고 하자.

**보조정리 1.** 하이퍼-토러스에서 출발노드에서 목적노드로 라우팅할 경우 두 개의 외부 예지를 이어주는 기본모듈의 내부 예지 개수는 기본모듈 내의 두 노드 사이의 거리이고 그 거리는 다음과 같다.

(경우 1) 내부 경로 길이가 1인 경우

- 1) 가로 예지  $((x-1,y,111), (x,y,011))$ 와 가로 예지  $((x,y,111), (x+1,y,011))$
- 2) 세로 예지  $((x,y-1,101), (x,y,001))$ 와 세로 예지  $((x,y,101), (x,y+1,001))$
- 3) 사선 예지  $((x-1,y-1,110), (x,y,010))$ 와 사선 예지  $((x,y,110), (x+1,y+1,010))$
- 4) 역사선 예지  $((x-1,y+1,100), (x,y,000))$ 와 역사선 예지  $((x,y,100), (x+1,y-1,000))$
- 5) 가로 예지  $((x-1,y,011), (x,y,111))$ 와 세로 예지  $((x,y,101), (x,y+1,001))$
- 6) 가로 예지  $((x-1,y,111), (x,y,011))$ 와 세로 예지  $((x,y,001), (x,y-1,101))$
- 7) 사선 예지  $((x-1,y-1,110), (x,y,010))$ 와 역사선 예지  $((x,y,000), (x-1,y+1,100))$
- 8) 사선 예지  $((x+1,y+1,010), (x,y,110))$ 와 역사선 예지  $((x,y,100), (x+1,y-1,000))$
- 9) 세로 예지  $((x,y-1,101), (x,y,001))$ 와 역사선 예지  $((x,y,000), (x-1,y+1,100))$
- 10) 세로 예지  $((x,y+1,001), (x,y,101))$ 와 역사선 예지  $((x,y,100), (x+1,y-1,000))$

(경우 2) 내부 경로 길이가 2인 경우

- 11) 가로 예지  $((x+1,y,011), (x,y,111))$ 와 세로 예지  $((x,y,001), (x,y-1,101))$
- 12) 가로 예지  $((x-1,y,111), (x,y,011))$ 와 세로 예지  $((x,y,101), (x,y+1,001))$
- 13) 가로 예지  $((x-1,y,111), (x,y,011))$ 와 사선 예지  $((x,y,110), (x+1,y+1,010))$
- 14) 가로 예지  $((x+1,y,011), (x,y,111))$ 와 사선 예지  $((x,y,010), (x-1,y-1,110))$
- 15) 사선 예지  $((x+1,y+1,010), (x,y,110))$ 와 역사선 예지  $((x,y,000), (x-1,y+1,100))$
- 16) 사선 예지  $((x-1,y-1,110), (x,y,010))$ 와 역사선 예지  $((x,y,100), (x+1,y-1,000))$

(경우 3) 내부 경로 길이가 3인 경우

- 17) 세로 예지  $((x,y-1,101), (x,y,001))$ 와 사선 예지  $((x,y,110), (x+1,y+1,010))$
- 18) 세로 예지  $((x,y+1,001), (x,y,101))$ 와 사선 예지  $((x,y,010), (x-1,y-1,110))$

19) 가로 에지  $((x+1,y,011), (x,y,111))$ 와 역사선 에지  $((x,y,000), (x-1,y+1,100))$

20) 가로 에지  $((x-1,y,111), (x,y,011))$ 와 역사선 에지  $((x,y,100), (x+1,y-1,000))$

**증명.** 그림 5는 하이퍼-토러스의 기본모듈인 3차원 하이퍼큐브를 보여주는데, 노드 011, 111은 가로 에지, 노드 001, 101은 세로 에지, 노드 010, 110은 사선 에지, 그리고 노드 000, 100은 역사선 에지와 결합해 있다. 가로 에지와 결합된 두 노드 011과 111은 단지 1비트 다르기 때문에 두 노드 사이의 거리는 1이고, 연속된 가로 에지 사이의 기본모듈에서의 내부 경로길이는 1이다. 이와 동일하게 세로 에지, 사선 에지, 역사선 에지에 결합된 두 노드는 모두 1비트 다르기 때문에 연속된 세로 에지, 사선 에지, 역사선 에지 사이의 기본모듈에서의 내부 경로길이는 1이다. 임의의 두 노드 사이의 라우팅에서 지나쳤던 경로를 되돌아오는 경우는 없기 때문에 두 외부 에지 사이의 내각이 90도 미만인 경우는 제외하였고, 90도 초과하는 에지는 5개이다. 여기에서 기본모듈을 하나의 점으로 볼 때 하나의 점에서 연결된 8개의 외부에지 중 두 개의 에지가 이루는 각에서 적은 각을 내각이라 하고 큰 각을 외각이라고 한다. 총 8개의 에지에 결합된 노드에서 각각 5개의 에지(90도 초과하는)에 결합된 노드로 라우팅하는 경우를 생각해볼 수 있으므로 40개의 경우의 수를 생각할 수 있는데 양방향 링크이므로 20개의 경우를 보조정리 1에 정리 하였다. 위에서 증명한 4개의 연속된 에지를 제외한 나머지 경우는 그림 5로 충분하기 때문에 생략하고, 보조정리 1에서 내부 경로길이 1의 2)와 내부 경로길이 2의 13)을 그림 5에 표시하였다.

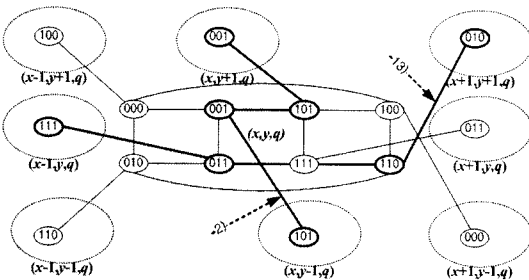


그림 5 하이퍼-토러스의 기본모듈과 에지

### 3.2 라우팅 알고리즘과 성질

#### 3.2.1 단순 라우팅 알고리즘

하이퍼-토러스에서 단순 라우팅 알고리즘은 출발 노드가 있는 출발 기본모듈에서 목적 노드가 있는 목적 기본모듈로 라우팅하고, 목적 기본모듈 내에서 목적 노드로 라우팅 한다. 출발 기본모듈과 목적 기본모듈의 노

드 조합은 고려하지 않는다. 같은 기본모듈 내에서 노드 간의 라우팅은 내부 라우팅이라 하고, 다른 기본모듈에 있는 노드간의 라우팅을 외부 라우팅이라 한다. 단순 라우팅의 기본 전략은 x축과 y축으로 각각 1만큼 떨어져 있는 기본모듈로 라우팅 할 때 가로 에지나 세로 에지를 이용하지 않고 사선 에지를 이용한다. 노드  $U(x_1, y_1, q)$ 를 출발 노드,  $V(x_2, y_2, q')$ 를 목적 노드라고 하자.  $dx$ 는 x 좌표가 증가하는 방향으로 x축 거리,  $dy$ 는 y 좌표가 증가하는 y축 거리,  $dm$ 은 x축과 y축을 동시에 움직이는 거리,  $dr$ 은  $dm$ 만큼 이동한 후 x축이나 y축으로 남은 거리를 나타낸다.  $\rightarrow$ 는 내부 라우팅을  $\Rightarrow$ 는 외부 라우팅을 표시한다.  $dx=(x_2-x_1+m)/m$ 이고,  $dy=(y_2-y_1+n)/n$ 이며,  $dm=\min(dx,dy)$ 이고,  $dr=\max(dx,dy) - \min(dx,dy)$ 이다.

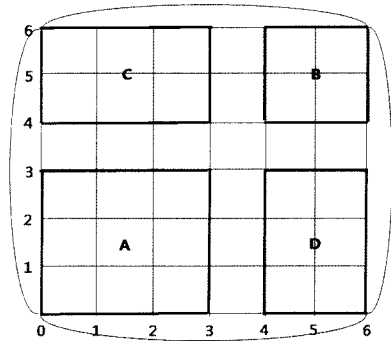


그림 6 QT(7,7) 라우팅 영역

단순 라우팅 알고리즘은  $dx$ 와  $dy$ 에 따라서 다음과 같이 4가지 경우가 있다. 출발 노드  $U$ 가 그림 6처럼 기본모듈  $(0,0)$ 에 있다면 목적 노드  $V$ 는  $dx \leq \lfloor \frac{m}{2} \rfloor$ ,  $dy \leq \lfloor \frac{n}{2} \rfloor$  이면 A영역에 있고,  $dx > \lfloor \frac{m}{2} \rfloor$ ,  $dy > \lfloor \frac{n}{2} \rfloor$  이면 B영역에 있고,  $dx \leq \lfloor \frac{m}{2} \rfloor$ ,  $dy > \lfloor \frac{n}{2} \rfloor$  이면 C영역에 있고,  $dx > \lfloor \frac{m}{2} \rfloor$ ,  $dy \leq \lfloor \frac{n}{2} \rfloor$  이면 D영역에 있다. 목적 노드  $V$ 가 B영역에 있다면 xy축에 대칭 이동시키고, C영역에 있다면 x축에 대칭, D영역에 있다면 y축에 대칭 이동시키면 A영역과 동일하다. 즉, A영역의 라우팅 알고리즘에 대해서 C영역은 x축, D영역은 y축, B영역은 역사선 45도의 xy축에 대칭하여 라우팅하면 되므로, 라우팅 알고리즘은 A영역에 대해서만 다룬다. 그림 6의 A영역에서 x축으로 가장 먼 거리는  $\lfloor \frac{m}{2} \rfloor$ , y축으로 가장 먼 거리는  $\lfloor \frac{n}{2} \rfloor$ 이다. 사선 에지를 이용

하면 x축과 y축을 동시에 1씩 증감시키며 이동할 수 있으므로 보조정리 2와 같은 결과를 가질 수 있다.

**보조정리 2.**  $QT(m,n)$ 의 외부 에지 최대사용횟수는  $\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)$ 이다.

단순 라우팅 알고리즘은 다음과 같다.

**[단계 1]** 출발 기본모듈 내부 라우팅

$dm > 0$  이면, 출발 노드  $U(x_1, y_1, q)$ 에서 사선 에지에 결합된 노드  $(x_1, y_1, 110)$ 로 라우팅 하고 [단계 2]로 이동.

$dm = 0$ 이고  $dx > 0$  이면, 출발 노드  $U(x_1, y_1, q)$ 에서 가로 에지에 결합된 노드  $(x_1, y_1, 111)$ 로 라우팅 하고 [단계 3]으로 이동.

$dm = 0$ 이고  $dy > 0$ 이면, 출발 노드  $U(x_1, y_1, q)$ 에서 세로 에지에 결합된 노드  $(x_1, y_1, 101)$ 로 라우팅 하고 [단계 3]으로 이동.

**[단계 2]** 사선 에지를 이용한 외부 라우팅

노드  $(x_1, y_1, 110) \Rightarrow$  노드  $(x_1+1, y_1+1, 010) \rightarrow$  노드  $(x_1+1, y_1+1, 110) \Rightarrow$  노드  $(x_1+2, y_1+2, 010)$ 의 과정을  $dm$ 만큼 반복한다.

**[단계 3]** 가로(세로) 에지를 이용한 외부 라우팅

$dx > dy$ 이면, 노드  $(x_1+dm, y_1+dm, 111) \Rightarrow$  노드  $(x_1+dm+1, y_1+dm+1, 011) \rightarrow$  노드  $(x_1+dm+1, y_1+dm+1, 111) \Rightarrow$  노드  $(x_1+dm+2, y_1+dm+2, 011)$ 의 과정을  $dr$ 만큼 반복.

$dy > dx$  이면, 노드  $(x_1+dm, y_1+dm, 101) \Rightarrow$  노드  $(x_1+dm+1, y_1+dm+1, 001) \rightarrow$  노드  $(x_1+dm+1, y_1+dm+1, 101) \Rightarrow$  노드  $(x_1+dm+2, y_1+dm+2, 001)$ 의 과정을  $dr$ 만큼 반복.

**[단계 4]** 목적 기본모듈의 내부 라우팅

노드  $\{(x_2, y_2, 010), (x_2, y_2, 011), (x_2, y_2, 001)\}$ 로부터 목적 노드  $V(x_2, y_2, q')$ 로 라우팅 한다.

예를 들어,  $QT(7,7)$ 에서  $U=(0,0,101)$ ,  $V=(4,3,001)$ 인 경우 라우팅 경로는 다음과 같다.  $U=(0,0,101) \rightarrow (0,0,100) \rightarrow (0,0,110) \Rightarrow (1,1,010) \rightarrow (1,1,110) \Rightarrow (2,2,010) \rightarrow (2,2,110) \Rightarrow (3,3,010) \rightarrow (3,3,011) \rightarrow (3,3,111) \Rightarrow (4,3,011) \rightarrow (4,3,001)=V$ .

3.2.2 최적 라우팅 알고리즘

최적 라우팅 알고리즘은 단순 라우팅 알고리즘을 활용하지만, 단순 라우팅 알고리즘에서 고려하지 않았던 출발 기본모듈과 목적 기본모듈에서의 노드 조합을 고려하였다. 이것은 출발 기본모듈과 목적 기본모듈에서 라우팅에 사용되는 내부에지의 수(내부 경로길이)를 최소화한다. 최적 라우팅 알고리즘은 다음과 같다.

**[단계 1]** 출발 기본모듈의 내부 라우팅

단순 라우팅 알고리즘의 [단계 1]과 같다.

**[단계 2]** 외부 라우팅

출발 기본모듈에서 노드와 목적 기본모듈에서 노드 조합에 따라 단순 라우팅 알고리즘의 [단계 2]와 [단계 3]의 순서를 바꾸어 라우팅 한다.

**[단계 3]** 목적 기본모듈의 내부 라우팅

단순 라우팅 알고리즘의 [단계 4]와 같다.

최적 라우팅 알고리즘에서는 [단계 2]만을 다룬다.

**[단계 2]** 외부 라우팅

외부 라우팅은  $dx-dy$ 의 값에 따라서 다르다.  $dx-dy$ 의 값이 0인 경우, 0보다 큰 경우, 0보다 작은 경우로 나누어 생각할 수 있다. 0인 경우는 x축과 y축의 거리가 같고, 0보다 큰 경우는 x축이 y축에 거리가 길며, 0보다 적은 경우는 x축이 y축에 비해 거리가 짧은 경우이다. 라우팅에서 출발 기본모듈을  $U$ 라고 하고 출발 노드의 하이퍼큐브 주소를  $q$ 라고 하고 간단하게 출발 노드라고 한다. 목적 기본모듈을  $V$ 라고 하고 목적 노드의 하이퍼큐브 주소를  $q'$ 라고 하고 간단하게 목적 노드라고 한다. 이하 본 논문에서 나타내는 하이퍼큐브의 주소에서 x는 don't care를 뜻하고 0x0과 {000,010}은 같다. 라우팅의 방향은 그림 6의 A, B, C, D 영역과 같고 B, C, D 영역은 각각 A 영역에 대칭되므로 A 영역에 대해서만 다룬다.  $dx$ 와  $dy$ 의 값에 따라 다음과 같은 경우로 나누어 볼 수 있다.

**보조정리 3.** 하이퍼-토러스의 외부 라우팅은 다음과 같은 경우들로 나누어 볼 수 있다.

(경우 1)  $dx-dy=0$ 인 경우

그림 7에서, A 경로로 라우팅 한다.

(경우 2.1)  $dx-dy > 0$ 이고  $dx-dy$ 가 홀수인 경우

그림 8에서,  $q=\{xx0\}$ 이면 A 경로,  $q=\{xx1\}$ 이면 B 경로로 라우팅 한다.

(경우 2.2)  $dx-dy > 0$ 이고  $dx-dy$ 가 짝수인 경우

그림 9에서,  $q=\{x11\}$ 이고  $q'=\{x1x\}$ 이면 B 경로,  $q=\{x0x\}$ 이고  $q'=\{x1x\}$ 이면 C 경로,  $q=\{110\}$ 이고  $q'=\{x11\}$ 이면 D 경로, 그리고 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다.

(경우 3.1)  $dy-dx > 0$ 이고  $dx=0$ 인 경우

그림 10에서,  $q=\{010\}$ 이고  $q'=\{00x\}$ 이면 B 경로,  $q=\{001\}$ 이고  $q'=\{x0x\}$ 이거나,  $q=\{011,100\}$ 이고  $q'=\{00x\}$ 이거나,  $q=\{101\}$ 이고  $q'=\{xxx\}$ 이거나, 그리고  $q=\{11x\}$ 이고  $q'=\{0xx\}$ 이면 C 경로,  $q=\{000\}$ 이고  $q'=\{xxx\}$ 이거나,  $q=\{001,010,011,100\}$ 이고  $q'=\{x1x\}$ 이면 D 경로, 그리고 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다. 목적 노드가  $V$ 인 경우는 F 경로와 G 경로만으로 라우팅한다.

(경우 3.2)  $dy-dx > 0$ 이고  $dy-dx=1$ 인 경우

그림 11에서,  $q=\{x11\}$ 이고  $q'=\{1xx\}$ 이면 B 경로,  $q=\{x01\}$ 이고  $q'=\{xxx\}$ 이거나,  $q=\{011\}$ 이고  $q'=\{011,$

100)이거나,  $q=\{100\}$ 이고  $q'=\{x1x\}$ 이거나,  $q=\{111\}$ 이고  $q'=\{010, 001\}$ 이면 C 경로,  $q=\{000\}$ 이면  $q'=\{xxx\}$ 이면 D 경로, 그리고 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다.

(경우 3.3)  $dy-dx>0$ 이고 3.1과 3.2를 제외한 경우

그림 12에서, 목적노드가 V일 경우,  $q=\{00x\}$ 이고  $q'=\{01x\}$ 이면 C 경로로 라우팅 하고, 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다. 목적노드가 V'일 경우에는 G 경로로만 라우팅한다.

**증명.** [단계 1]과 [단계 3]은 단순라우팅에 따라 라우팅하므로 증명은 필요 없다. 다음은 최적라우팅 중 [단계 2] 외부 라우팅에 대한 증명이다.

출발 기본모듈과 목적 기본모듈에서의 라우팅을 제외한 라우팅을 외부 라우팅이라 하고 그 길이를 외부 길이라고 한다. 출발 기본모듈과 목적 기본모듈에서의 라우팅을 내부 라우팅이라고 하고 그 길이를 내부 길이라고 한다. 그림 7, 8, 9, 10, 11, 12에서 알파벳 A, B, C, D, E, F, G, H는 고려해볼 수 있는 라우팅 경로를 의미하고, 라우팅 경로가 꺾인 부분의 숫자는 방향 전환시 중간경유 기본모듈의 내부 경로길이이고 보조정리 1에 정리되어 있다.

(경우 1)

그림 7에서, 경로 A의 외부 길이는 외부 에지의 수  $dx$ 와 중간경유 기본모듈의 내부 에지의 수  $dx-1$ 을 합한  $2dx-1$ 이다. 경로 B, C의 외부 길이는 외부 에지의 수  $dx+1$ 과 중간경유 기본모듈의 내부 에지의 수  $dx+3$ 을 합한  $2dx+4$ 이다. 경로 A는 B, C 경로보다 외부 길이가 5 짧다.

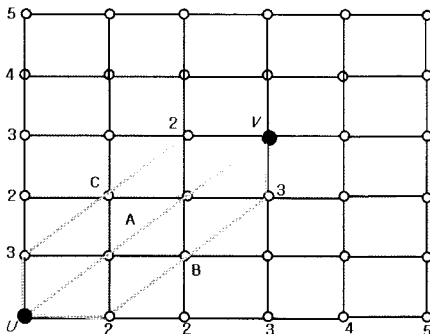


그림 7  $dx=dy$ 인 경우 라우팅 경로

출발 기본모듈 U에서 A 경로의 사선 에지와 결합된 노드 110은 B 경로의 가로 에지와 결합된 노드 111과 1의 거리에 있고, C 경로의 세로 에지와 결합된 노드 101과 2의 거리에 있다. 출발 노드 q에서 노드 110까지의 거리는 노드 111까지의 거리보다 최대 1 크고, 노드

101까지의 거리보다 최대 2 크다. 따라서 출발 기본모듈에서 A 경로의 내부 길이는 B 경로의 내부 길이 보다 최대 1 크고 C 경로의 내부 길이보다 최대 2 크다. 목적 기본모듈 V에서 A 경로의 사선 에지와 결합된 노드 010은 B 경로의 세로 에지와 결합된 노드 001과 2의 거리에 있고, C 경로의 가로 에지와 결합된 노드 011과 1의 거리에 있다. 목적 노드 q'에서 노드 010까지의 거리는 노드 001까지의 거리보다 최대 2 크고, 노드 011까지의 거리보다 최대 1 크다. 따라서 목적 기본모듈에서 A 경로의 내부 길이는 B 경로의 내부 길이 보다 최대 2 크고 C 경로의 내부 길이보다 최대 1 크다.

경로 A의 외부 길이는 B, C 경로의 외부 길이보다 5가 짧고, A 경로의 내부 길이는 B, C 경로의 내부길이보다 출발 기본모듈과 목적 기본모듈에서 최대한 3이 크므로 A 경로의 라우팅 길이는 B, C 경로의 라우팅 길이보다 최소한 2가 짧다. 따라서 어떤 경우라도 A 경로로 라우팅한다.

그림 7에서 격자상의 숫자는 보조정리 1에서 정의한 라우팅 경로에서 방향전환시 기본모듈 내의 내부 경로 길이로 이것은 그림 12에까지 같은 의미로 사용된다.

(경우 2.1)

그림 8에서, (경우 1)에 의해서 A 경로는 C 경로보다 라우팅 길이가 짧고, B 경로는 D 경로보다 라우팅 길이가 짧다. 따라서 C, D 경로는 라우팅 경로에서 제외된다.  $dx-dy>0$ 에서  $dx>dy$  이므로, A 경로와 B 경로의 외부 길이는  $2dx$ 로 같다.

출발 기본모듈 U에서 A 경로의 사선 에지와 결합된 노드 110은 B 경로의 가로 에지와 결합된 노드 111과 1의 거리에 있다. 목적 기본모듈 V에서 A 경로의 가로 에지와 결합된 노드 011은 B 경로의 사선 에지와 결합된 노드 010과 1의 거리에 있다. A, B 경로 중 어떤 경로를 선택하더라도 선택한 경로가 출발 기본모듈에서 내부 길이가 1만큼 작다면 선택하지 않는 경로보다 내부 길이가 같거나 짧다. 따라서  $q$ 가  $\{xx0\}$ 이면 경로 A로 라우팅하고,  $q$ 가  $\{xx1\}$ 이면 경로 B로 라우팅한다.

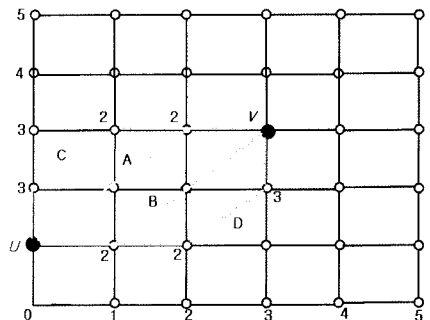


그림 8  $dx-dy>0$ 이고  $dx-dy$ 가 홀수인 경우 라우팅 경로

(경우 2.2)

그림 9에서, B 경로의 외부 길이는 외부 에지의 수  $dx$ 와 중간경유 기본모듈의 내부 에지의 수  $dx-2+2$ 를 합한  $2dx$ 이다. E 경로의 외부 길이는 외부 에지의 수  $dx+dy$ 와 중간경유 기본모듈의 내부 에지의 수  $dx-1+dy-1+2$ 를 합한  $2(dx+dy)$ 이다.

출발 기본모듈  $U$ 에서 B 경로의 가로 에지와 결합된 노드와 E 경로의 가로 에지와 결합된 노드는 모두 111이므로 출발 기본모듈에서 B, E 경로의 내부 길이의 차이는 없다. 목적 기본모듈  $V$ 에서 B 경로의 사선 에지와 결합된 노드 010과 E 경로의 세로 에지에 결합된 001은 2비트 차이가 있다. 따라서 목적 기본모듈에서 B 경로의 내부 길이는 최대한 2가 길다.

B 경로의 외부 길이는 E 경로의 외부 길이보다  $2dy$ 만큼 짧고, B 경로의 내부 길이는 E 경로의 내부 길이보다 목적 기본모듈에서 최대한 2 길므로,  $dy>0$  이면 A 경로의 라우팅 길이는 E 경로의 라우팅 길이보다 최소한 같거나 짧다. 마찬가지로 D 경로의 라우팅 길이는 F 경로의 라우팅 길이보다 최소한 같거나 짧다.

그림 9에서 A, B, C, D 경로의 외부 길이는 같다. 출발 기본모듈  $U$ 에서 A 경로의 사선 에지와 결합된 노드와 D 경로의 사선 에지와 결합된 노드는 모두 110이므로 출발 기본모듈에서 A, D 경로의 내부 길이의 차이는 없다. 목적 기본모듈  $V$ 에서 A 경로의 역사선 에지와 결합된 노드 000과 D 경로의 가로 에지에 결합된 011은 2 비트 차이가 있다.  $q'$ 가 000의 보수 111이면 D 경로를,  $q'$ 가 011의 보수인 100이면 A 경로를,  $q'$ 가 그 외의 노드일 경우 어떤 경로를 선택하여도 목적 노드에서 내부 길이는 같다. 따라서  $q'$ 가 {011,111}이면 D 경로로 라우팅하고 그렇지 않다면 A 경로로 라우팅한다. 마찬가지로 목적 기본모듈에서 C, B 경로의 내부 길이는 같고, 출발 기본모듈에서 C 경로의 사선 에지와 결합된 100과 B 경로의 가로 에지와 결합된 노드 111은 2 비트 차이가 있으므로  $q$ 가 {000,100}이면 C 경로로 라우팅하고 그렇지 않다면 B 경로로 라우팅한다.

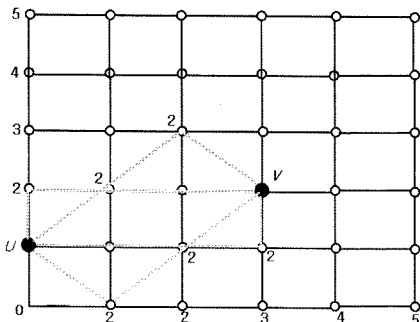


그림 9  $dx-dy>0$ 이고  $dx-dy$ 가 짝수인 경우 라우팅 경로

(경우 3.1)

그림 10에서, 목적 노드가  $V$ 일 경우  $dy$ 가 짝수일 경우로 A, B, C, D 경로는 기본모듈간을 연결하는 외부 에지  $dy$ , 기본모듈 내의 경로 길이  $dy-1$ 이고, E 경로는 기본모듈간을 연결하는 외부에지  $dy$ , 기본모듈 내의 경로길이  $dy-2+3$ 이다. 따라서 A, B, C, D 경로의 외부 길이는  $2dy-1$ 이고, E 경로의 외부 길이는  $2dy+1$ 이므로 A, B, C, D 경로가 E 경로보다 외부 길이가 2가 짧다. E 경로는 출발 기본모듈에서 D 경로와 결합된 노드 000과 같고 도착 기본모듈에서 D 경로와 결합된 노드 010과 2비트 차이이다. E 경로의 경우 어떠한 경우에도 D 경로와 같거나 길어지기 때문에 라우팅에서 제외된다. A, B, C, D 경로의 경우 출발 기본모듈의 출발 노드  $q$ 와 목적 기본 모듈의 목적 노드  $q'$ 의 위치에 따라서 라우팅의 경로가 정해진다.  $q=(010)$ 이고  $q'=(00x)$ 이면 B 경로,  $q=(001)$ 이고  $q'=(x0x)$ ,  $q=(011,100)$ 이고  $q'=(00x)$ ,  $q=(101)$ 이고  $q'=(xxx)$ , 그리고  $q=(11x)$ 이고  $q'=(0xx)$ 이면 C 경로,  $q=(000)$ 이고  $q'=(xxx)$ ,  $q=(001,010,011,100)$ 이고  $q'=(x1x)$ 이면 D 경로, 그리고 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다. 그림 10에서, 목적 노드가  $V'$ 일 경우는  $dy$ 가 홀수일 경우로 출발 기본모듈 내에서 B 경로와 결합된 노드 110과 C 경로와 결합된 노드 101은 2비트 차이가 나나 목적 기본 모듈 내에서 결합된 노드는 001로 같다. 따라서  $q=(010,110)$ 이고  $q'=(xxx)$ 이면 A, B 경로를 연장한 F 경로로 라우팅하고 나머지  $q$ 와  $q'$ 의 노드들은 C 경로를 연장한 G 경로로 라우팅한다.

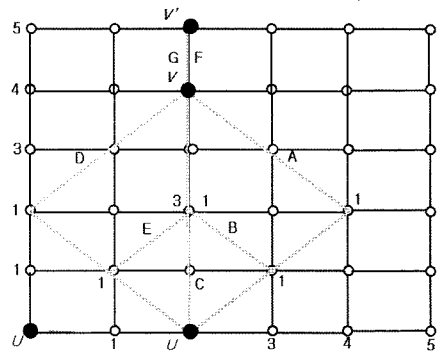


그림 10  $dx-dy<0$ 이고  $dx=0$ 인 경우 라우팅 경로

(경우 3.2)

그림 11에서, A, C 경로는 기본모듈 간을 연결하는 외부 에지  $dy$ , 기본모듈 내의 경로 길이  $dy-2+3$ 이고, B, D 경로는 기본모듈 간을 연결하는 외부 에지  $dy+1$ , 기본모듈 내의 경로 길이  $dy+1-2+2$ 이다. 따라서 A, C 경로의 외부 길이는  $2dy+1$ 이고 B, D 경로의 외부 길이



는  $2dy+2$ 이므로 A, C 경로는 B, D 경로보다 외부 길이는 1이 짧다. 출발 기본모듈에서 A 경로와 결합된 노드 110과 B 경로와 결합된 노드 111은 1 비트 차이이고, C 경로와 결합된 노드 101과 D 경로와 결합된 노드 000은 2 비트 차이이다. 목적 기본모듈에서 A 경로와 결합된 노드 001과 B 경로와 결합된 노드 100은 2 비트 차이이고, C 경로와 결합된 노드 010과 D 경로와 결합된 노드 011은 1 비트 차이이다. 출발 기본모듈의 출발 노드  $q$ 와 목적 기본모듈의 목적 노드  $q'$ 의 위치에 따라서 라우팅의 경로가 정해진다.  $q=\{x11\}$ 이고  $q'=\{1xx\}$ 이면 B 경로,  $q=\{x01\}$ 이고  $q'$ 가 모든 노드이거나,  $q=\{011\}$ 이고  $q'=\{011, 100\}$ 이거나,  $q=\{100\}$ 이고  $q'=\{x1x\}$ 이거나,  $q=\{111\}$ 이고  $q'=\{010, 001\}$ 이면 C 경로,  $q=\{000\}$ 이면  $q'$ 에 상관없이 D 경로, 그리고 나머지  $q$ 와  $q'$ 의 노드들은 A 경로로 라우팅한다.

(경우 3.3)

그림 12에서, 목적 노드가 V일 경우  $dy-dx$ 가 0을 제외한 짝수일 경우로 A 경로의 외부 길이는 외부 에지의 수  $dy$ 와 중간경유 기본모듈의 내부 에지의 수  $dy-1$ 을 합한  $2dy-1$ 이다. E 경로의 외부 길이는 외부 에지의 수  $dy+1$ 과 중간경유 기본모듈의 내부 에지의 수  $dy+2$ 를 합한  $2dy+3$ 이다. B 경로의 외부 길이는 외부 에지의 수  $dy-1$ 과 중간경유 기본모듈의 내부 에지의 수  $dy-1+3$ 을 합한  $2dy+1$ 이다. A 경로는 B 경로보다 외부 길이가 2 적고, E 경로보다 외부 길이는 4가 짧다.

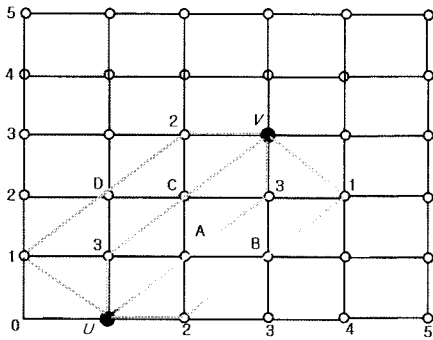


그림 11  $dx-dy < 0$ 이고  $dy-dx=1$ 인 경우 라우팅 경로

출발 기본모듈 U에서 A 경로의 사선 에지와 결합된 노드 110은 E 경로의 가로 에지와 결합된 노드 111과 1의 거리에 있고, B 경로의 사선 에지와 결합된 노드와 같다. 출발 노드  $q$ 에서 노드 110까지의 거리는 노드 111까지의 거리보다 최대한 1이 길다. 따라서 출발 기본모듈에서 A 경로의 내부 길이는 E 경로의 내부 길이보다 최대한 1이 길고 B 경로의 내부 길이와는 같다. 목적 기본모듈 V에서 A 경로의 역사선 에지와 결합된

노드 100은 E, B 경로의 세로 에지와 결합된 노드 001과 2의 거리에 있다. 목적 노드  $q'$ 에서 노드 100까지의 거리는 노드 001까지의 거리보다 최대한 2가 길다. 따라서 목적 기본모듈에서 A 경로의 내부 길이는 E, B 경로의 내부 길이보다 최대한 2가 길다.

A 경로의 외부 길이는 B 경로의 외부 길이보다 2가 짧고, A 경로의 내부 길이는 B 경로의 내부 길이보다 출발 기본모듈과 목적 기본모듈에서 최대한 2가 길므로 A 경로의 라우팅 길이는 B 경로의 라우팅 길이보다 최소한 같거나 짧다. A 경로의 외부 길이는 E 경로의 외부 길이보다 4가 짧고, A 경로의 내부 길이는 E 경로의 내부 길이보다 출발 기본모듈과 목적 기본모듈에서 최대한 3이 길므로 A 경로의 라우팅 길이는 E 경로의 라우팅 길이보다 최소한 1이 짧다. 따라서 B, E 경로는 라우팅에서 사용되지 않는다. 같은 방법으로 C 경로와 D, F 경로를 비교하면 D, F 경로는 라우팅에서 사용되지 않는다.

A 경로와 C 경로의 외부 길이는  $2dy-1$ 로 같다. 출발 기본모듈 U에서 A 경로의 사선 에지와 결합된 노드 110은 C 경로의 역사선 에지와 결합된 노드 000과 2의 거리에 있고 목적 기본모듈 V에서 A 경로의 역사선 에지와 결합된 노드 100은 C 경로의 사선 에지와 결합된 노드 010과 2의 거리에 있다.  $q$ 가 110의 보수 001이면 C 경로를 선택하고,  $q$ 가 000의 보수 111이면 A 경로를 선택하고,  $q$ 가 그 이외의 노드들일 경우 어떤 경로를 선택하여도 출발 노드에서 내부 길이는 같다.  $q'$ 가 100의 보수 011이면 C 경로를,  $q'$ 가 010의 보수인 101이면 A 경로를,  $q'$ 가 그 외의 노드일 경우 어떤 경로를 선택하여도 목적 노드에서 내부 길이는 같다. 따라서  $q$ 가  $\{000, 001\}=\{00x\}$ 이거나  $q'$ 가  $\{010, 011\}=\{01x\}$ 이면 C 경로로 라우팅 하고 그렇지 않다면 A 경로로 라우팅 한다.

그림 12에서 목적 노드가 V'일 경우  $dy-dx$ 가 1을 제외한 홀수일 경우로 A 경로를 연장한 G 경로는 기본모

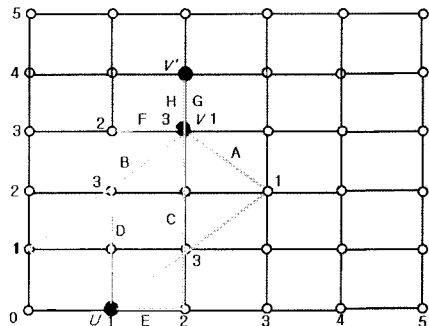


그림 12  $dx-dy < 0$ 이고 3.1과 3.2를 제외한 경우 라우팅 경로

들 간을 연결하는 외부에서  $dy$ , 기본모듈 내의 경로 길이  $dy-1$ 이고, B 경로를 연장한 H 경로는 기본모듈 간을 연결하는 외부에서  $dy$ , 기본모듈 내의 경로 길이  $dy-2+3$ 이다. 따라서 G 경로의 외부 길이는  $2dy-1$ 이고, H 경로의 외부 길이는  $dy+1$ 이므로 G 경로가 H 경로보다 2비트 길다. 출발 기본모듈에서 G 경로와 결합된 노드 110과 H 경로와 결합된 노드 000은 2비트 차이가 남으로 H 경로가 G 경로의 길이 보다 어떤 경우든 같거나 길다. 따라서 목적 노드가  $V'$ 일 경우는 G 경로로만 라우팅한다.

3.2.3 지름

지름은 내부 경로 길이와 외부 경로 길이로 나누어서 분석한다. 보조정리 2에 의해서 외부 경로 길이는  $\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)$ 이다. 기본모듈의 내부 경로 길이는 3차원 하이퍼큐브의 지름이 3이기 때문에 최대 3이다. 출발 기본모듈, 목적 기본모듈,  $\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)-1$ 개의 중간경유 기본모듈에서 내부 경로가 있다. 출발(목적) 기본모듈의 내부 경로 길이는 최대 2이다. 중간경유 기본모듈의 내부 경로 길이는 라우팅에서 나타나는 에지의 연속된 순서에 따라 달라진다. 연속된 가로(세로) 에지와 사선(역사선)에지 사이의 중간경유 기본모듈 내부 경로 길이와 사선 에지와 사선 에지의 교차점에 있는 기본모듈 내부의 경로 길이는 1이다. 사선 에지와 가로 에지의 교차점에 있는 기본모듈의 내부 경로 길이, 가로 에지와 사선 에지의 교차점에 있는 기본모듈 내부의 경로 길이, 사선 에지와 역사선 에지와의 교차점에 있는 기본모듈 내부의 경로 길이, 사선 에지와 역사선 에지의 교차점에 있는 기본모듈 내부의 경로 길이, 그리고 가로 에지와 세로 에지의 교차점에 있는 기본모듈 내부의 경로 길이는 2이다. 사선 에지와 세로 에지의 교차점에 있는 기본모듈 내부의 경로 길이, 세로 에지와 사선 에지의 교차점에 있는 기본모듈 내부의 경로 길이는 3이다. 라우팅에 사용되는 에지는 사선(역사선)에지만으로 구성되거나, 가로(세로)에지만으로 구성되거나, 사선(역사선)에지-가로(세로)에지로 구성된다. 최악의 경우 x축으로  $\lfloor \frac{m}{2} \rfloor$ 만큼 떨어져 있거나, y축으로  $\lfloor \frac{n}{2} \rfloor$ 만큼 떨어진 노드로 라우팅하는 것이다. 이 경우 중간경유 기본모듈의 내부 경로 길이는  $\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)$ 이다. 외부 경로 길이 + 중간경유 기본모듈의 내부 경로 길이 + 출발 기본모듈의 내부 경로 길이 + 목적 기본모듈의 내부 경로 길이가 지름이다. 따라서 정리 1과 같은 결과 값이 나온다.

정리 1.  $QT(m,n)$ 의 지름은  $\text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor) + \text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor)-1 + 2 + 3$ 이다.

$m=n$ 일 때

지름  $k = \text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor) + \text{Max}(\lfloor \frac{m}{2} \rfloor, \lfloor \frac{n}{2} \rfloor) - 1 + 2 + 3 = 2(\lfloor \frac{n}{2} \rfloor) + 4 = n + 4$ 이다.  $n$ 이 홀수일 때 출발 기본모듈에서 출발 노드의 거리가 1이고 목적 기본모듈에서 목적 노드의 거리가 1이므로 거리의 차는 2가 된다. 이 때 기본모듈에서 반대로 돌아 이동할 경우 외부 에지와 기본모듈을 하나씩 더 거쳐 가야 하므로 거리는 같아진다.  $n$ 이 짝수일 때 어느 쪽으로 이동하느냐 하는 것은 출발 기본모듈에서 출발 노드와 목적 기본모듈에서 목적 노드에 따라서 달라지며 홀수일 때 지름에 영향을 미치지 않는다.

3.2.4 이분 폭

이분 폭은 연결망을 노드수가 같은 두 개의 망으로 분리하기 위해 제거해야할 최소의 에지 개수이다. 분리된 두 개의 망은 노드 수가 1개 차이보다 같거나 적어야 한다. 이분 폭이 적으면 적은 수의 에지 고장만으로 망이 쉽게 분리되어 재 역할을 할 수 없으므로 이분 폭은 클수록 좋다. x축의 값이  $i$ 인 노드  $(i,y,q)$ 들과  $i-1$ 인 노드  $(i-1,y,q)$ 들의 연결을 끊기 위해서 필요한 최소 에지 수는 사선 에지  $n$ 개, 역사선 에지  $n$ 개, 가로 에지  $n$ 개이다. 두 개의 망으로 분리하기 위해서는 x축의 값이  $(i + \lfloor \frac{m}{2} \rfloor, y, q)$ 인 노드들과  $(i-1 + \lfloor \frac{m}{2} \rfloor, y, q)$ 인 노드들도 분리되어야 한다. 이 노드들이 분리되기 위해서 필요한 에지 수 또한  $3n$ 이다.  $n$ 이 홀수인 경우, 하나의 기본모듈을 이등분하여야 하는데 이 때 제거해야할 에지개수는 4이다. 따라서 이때에는  $3n+1$ 이 된다.

그림 13은 6x6 하이퍼-토러스 망의 이분 폭을 나타내고 있다.

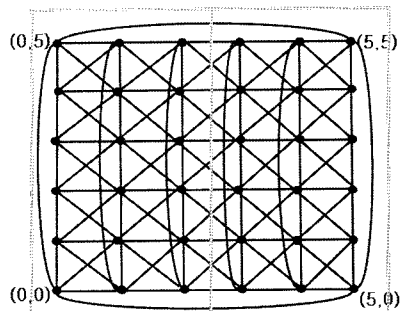


그림 13  $QT(5,5)$ 의 이분 폭

**정리 2.**  $QT(n,n)$ 의 이분 폭  $b$ 는  $n$ 이 짝수일 경우  $3n \times 2 = 6n$ 이다.  $n$ 이 홀수일 경우  $6n+1$ 이다.

3.2.5 다른 망들과의 비교

표 1은 분지수가 상수인 연결망이 같은 노드 개수를 가질 때 망 비용, 지름 그리고 이분 폭을 비교 하였다. 하이퍼-토러스 네트워크가 비교된 다른 연결망보다 지름과 망 비용은 적고 이분 폭은 크다. 표 1에서  $N$ 은 노드 개수이다[9].

3.2.6 해밀톤 사이클

상호연결망이 해밀톤 경로나 해밀톤 사이클 성질을 가지면 링이나 선형배열을 쉽게 구현할 수 있어 병렬처리에 유용하게 쓰이는 파이프라인으로 활용된다. 이분(bipartite) 그래프에서 임의의 흰색 정점과 임의의 검은색 정점 사이에 해밀톤 경로가 있으면 해밀톤 레이저블(laceable) 그래프라고 한다. 하이퍼큐브는 이분 그래프이고 해밀톤 레이저블 그래프이다[12]. 따라서 3차원 하이퍼큐브는 1이나 3의 거리에 있는 노드 사이에 해밀톤 경로가 존재한다. 그림 13의 화살표는 출발 노드 101과 1거리에 있는 목적 노드 001 사이의 해밀톤 경로를, 그림 14의 화살표는 출발 노드 001과 3거리에 있는 110 사이의 해밀톤 경로를 보여주고 있다.

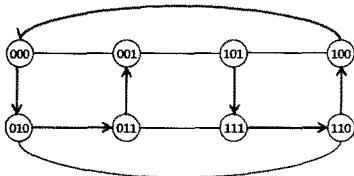


그림 14 거리 1에 있는 해밀톤 경로

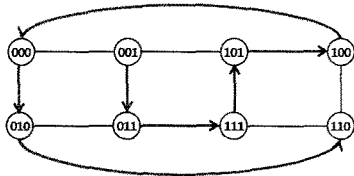


그림 15 거리 3에 있는 해밀톤 경로

하이퍼-토러스에서 해밀톤 사이클 알고리즘의 기본 개념은 다음과 같다. 기본모듈의 임의의 노드로부터 출발하여 그림 13과 그림 14의 경로를 따라 노드 101에 도착한다. 노드 101에서 자신을 제외한 모든 중간경유 기본모듈을 순회하고 출발 기본모듈의 노드 001에 도착한다. 노드 001로부터 그림 13과 그림 14의 경로에 따라 출발 노드에 도착한다. 출발 기본모듈을 제외한 모든 중간경유 기본모듈을 순회할 때 외부 에지는 세로 에지를 사용하고 상단의 가장자리 기본모듈에서는 사선 에지를 이용한다. 세로 에지와 사선 에지는 그림 13과 그림 14에 의해 해밀톤 경로가 존재하므로 중간경유 기본모듈에서도 해밀톤 경로가 존재한다. 다음은 하이퍼-토러스에서 해밀톤 사이클을 만드는 알고리즘이다.

- [단계 1] 출발 기본모듈에서 출발 노드로부터 노드 101까지 해밀톤 경로를 따라 이동
  - [단계 2] y축으로 마지막 행의 기본모듈까지 위쪽으로 이동
  - [단계 3] y축으로 마지막 행의 기본모듈이라면 그림 14처럼 해밀톤 경로를 따라 이동하여 사선 에지로 이동
  - [단계 4] 사선 에지로 들어온 후 해밀톤 경로를 따라 세로 에지로 이동
  - [단계 5] [단계 2]~[단계 4]를 반복
  - [단계 6] y축의 마지막 기본모듈에서 사선 에지로 이동하면 출발 기본모듈의 노드 010에 도착
  - [단계 7] 출발 기본모듈의 노드 010에서 하이퍼큐브 그래프의 해밀톤 경로를 따라 출발 노드에 도착
- 그림 15에서 화살표는 알고리즘에 따라 해밀톤 사이클을 보여주고 있다. 하이퍼-토러스는 그림 15와 같은 방법 이외에도 세로 에지를 이용하여 아래로 이동한 후 역사선 에지를 이용하여 이동하는 방법, 가로 에지를 이용하여 오른쪽에서 왼쪽으로 이동한 후 역사선 에지를 이용하여 이동하는 방법이 있다. 가로 에지를 이용하여 왼쪽에서 오른쪽으로 이동한 후 사선 에지를 이용하여 이동하는 방법은 존재하지 않는다. 이유는 보조 정리 1 (경우 2) 13)과 14)에서 보여주듯이 가로 에지를 이용하

표 1 연결망들의 망 비용과 이분 폭

연결망	망 척도	분지수	지름	망 비용	이분 폭
허니컴 메시		3	$1.63 \sqrt{N}$	$4.90 \sqrt{N}$	$0.82 \sqrt{N}$
다이아고널 메시		4	$\sqrt{N-1}$	$4 \sqrt{N-1}$	$4 \sqrt{N}$
토러스		4	$\sqrt{N}$	$4 \sqrt{N}$	$2 \sqrt{N}$
헥사고널 토러스		6	$0.58 \sqrt{N}$	$3.46 \sqrt{N}$	$4.61 \sqrt{N}$
허니컴 토러스		3	$0.81 \sqrt{N}$	$2.45 \sqrt{N}$	$2.04 \sqrt{N}$
피터슨-토러스		4	$0.47 \sqrt{N}$	$1.9 \sqrt{N}$	$0.1N + 1.58 \sqrt{N}$
하이퍼-토러스		4	$0.35 \sqrt{N} + 4$	$1.4 \sqrt{N} + 16$	$2.1 \sqrt{N}$

여 왼쪽에서 오른쪽으로 이동한 후 사선 에지로 이용하여 방향을 전환할 때와 가로 에지를 이용하여 오른쪽에서 왼쪽으로 이동한 후 역사선 에지를 이용하여 방향을 전환할 때 기본모듈 내에서 이동 경로의 길이가 2이기 때문이다.

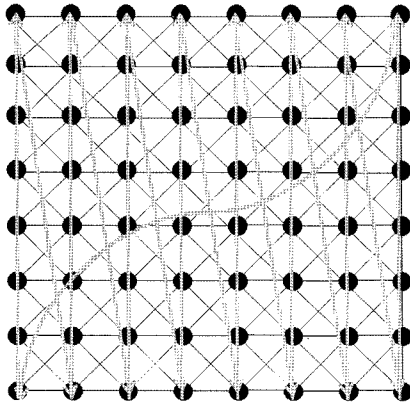


그림 16 하이퍼-토러스에서 해밀톤 사이클

#### 4. 결론

다중 컴퓨터의 성능은 다중 컴퓨터를 구성하는 각각의 프로세서와 프로세서 사이의 데이터 전송을 위한 통신 장비, 프로세서의 연결 구조를 제공하는 상호연결망 그리고 상호연결망에서 작성된 병렬 알고리즘에 좌우된다. 본 논문에서는 상호연결망의 확장성을 고려하여 분지수 4를 갖고, 효과적으로 병렬 알고리즘이 실행 가능한 3차원 하이퍼큐브를 기본모듈로 하는 하이퍼-토러스를 설계하였다. 하이퍼-토러스는 3차원 하이퍼큐브와 토러스 구조를 카티션곱 설계 방법과 유사하게 두 개의 그래프를 결합하는 방법이지만 분지수 4를 갖는다. 즉, 하이퍼-토러스는 카티션곱 방법에 의한 분지수보다 작은 수를 가지므로 망 확장과 망 비용 측면에서 효율적이다.

메시 부류의 상호연결망들은 노드수  $N$ 에 대해서 지름, 망 비용 그리고 이분 폭은  $O(\sqrt{N})$ 이다. 하이퍼-토러스의 망 비용  $1.4\sqrt{N}+16$ 은 토러스의 망 비용  $4\sqrt{N}$ 보다 약 3배 정도 향상되었고, 허니컴 토러스의 망 비용  $2.45\sqrt{N}$ 보다 약 2배 정도 개선된 값을 가진다. 이분 폭은 토러스와 허니컴 토러스와 비슷한 값을 가진다. 하이퍼-토러스는 3차원 하이퍼큐브와 토러스를 결합한 구조로써 하이퍼큐브와 토러스 구조에서 개발된 여러 가지 병렬 알고리즘의 효율적인 활용을 기대할 수 있는 연결망이다. 이를 위해 토러스 부류와 하이퍼큐브 부류의 임베딩에 관한 연구가 필요하겠다.

#### 참고 문헌

- [1] S. Abraham and K. Padmanabhan, "An Analysis of the Twisted Cube Topology," Proc. of the Int. Conference Parallel Processing, pp. 116-120, Aug. 1989.
- [2] S. B. Akers, D. Harel and B. Krishnamurthy, "The Star Graph: An Attractive Alternative to the n-Cube," Proc. of the Int. Conference Parallel Processing, pp. 393-400, 1987.
- [3] K. Efe, "A Variation on the hypercube with Lower Diameter," IEEE trans. computer, Vol.40, No.11, pp. 1312-1316, 1991.
- [4] B. Parhami and M. Rakov, "Perfect Difference Networks and Related Interconnection Structures for Parallel and Distributed Systems," IEEE trans. on Parallel and Distributed Systems, Vol.16, No.8, pp. 714-724, Aug. 2005.
- [5] C. D. Thompson and H. T. Kung, "Sorting on a Mesh-Connected Parallel Computer," Communications of the ACM, Vol.20 No.4, pp. 263-271, April 1977.
- [6] K. W. Tang and S. A. Padubidri, "Diagonal and toroidal Mesh Networks," IEEE trans. computer, Vol.43, No.7 pp. 815-826, Jul 1994.
- [7] M. S. Chen and K. G. Shin, "Addressing, Routing, and Broadcasting in Hexagonal Mesh Multiprocessors," IEEE trans. comput., Vol.39, No.1 pp. 10-18, Jan 1990.
- [8] I. Stojmenovic, "Honeycomb Network: Topological Properties and Communication Algorithms," IEEE trans. on Parallel and Distributed Systems, Vol.8, No.10, pp. 1036-1042, Oct 1997.
- [9] 서정현, 이형욱, 장문석, "멀티컴퓨팅 시스템을 위한 피터슨-토러스(PT) 네트워크", 정보과학회논문지 제35권 6호, pp. 263-272, 2008.
- [10] Y. Saad and M. H. Schultz, "Topological Properties of hypercubes," IEEE trans. Computer., Vol.37, No.7, pp. 867-872, July 1988.
- [11] A. El-Amawy and S. Latifi, "Properties and Performances of Folded hypercubes," IEEE trans. Parallel and Distributed System, Vol.2, No.1, pp. 31-42, 1991.
- [12] 주낙근, 임형석, "그래프 연산과 그레이 코드를 이용한 상호연결망의 설계", 정보과학회논문지, 제19권 1호, pp. 587-590, 1992.
- [13] C-H Yeh and E. Varvarigos, "Macro-Star Networks: Efficient Low-Degree Alternatives to Star Graphs for Large-Scale Parallel Architectures," Frontier'96, Symp. on the Frontiers of Massively Parallel Computation, Oct. 1996.
- [14] S. Latifi and P. K. Srimani, "Transposition Networks as a Class of Fault-Tolerant Robust Networks," IEEE trans. Comput., Vol.45, No.2, pp. 230-238, Feb. 1996.
- [15] Akl S. G. and Stojmenovic I., "Fundamental algo-

rithm for the star and pancake interconnection networks with applications to computational geometry," Networks, Vol.23, pp. 215-225, 1993.

- [16] Q. M. Malluhi and M. A. Bayoumi, "The Hierarchical hypercube: A New Interconnection Topology for Massively Parallel Systems," IEEE trans. Parallel and Distributed System, Vol.5, No.1, pp. 17-30, Jan. 1994.
- [17] D.-R. Duh, G.-H Chen and J.-F. Fang, "Algorithm and Properties of a New Two-level Network with Folded hypercubes as Basic Modules," IEEE trans. Parallel and Distributed System, Vol.6, No.7, pp. 714-723, 1995.
- [18] W. Shi and P. K. Srimani, "Hierarchical star: a new two level interconnection network," Journal of Systems Architecture, Vol.51, No.2, pp. 1-14, 2005.
- [19] Ohring and S. K. DAS, "Folded Petersen cube networks: new competitors for the hypercubes," IEEE trans. on Parallel and Distributed Systems, Vol.7, No.2, pp. 151-168, Feb 1996.
- [20] L. Wang, ZP. Chen and XH Jiang, "Ringed Petersen Spheres Connected hypercube Interconnection Networks," Proceedings of the 10th IEEE International Conference on Engineering of Complex Computer Systems (ICECCS'05), pp. 127-131, 2005.
- [21] S.-C. Ku, B.-F. Wang and T-K. Hung, "Constructing Edge-Disjoint Spanning Trees in Product Networks," IEEE Trans. Parallel and Distributed System, Vol.14, No.3, pp. 213-221, 2003.
- [22] C. Decayeux and D. Seme, "3D hexagonal Network: Modeling, Topological properties, Addressing Scheme, and Optimal Routing Algorithm," IEEE trans. Parallel and Distributed System, Vol.16, No.9, pp. 875-884, 2005.
- [23] A. Gonzalez, V. Garcia and L. D. Cerio, "Executing Algorithm with Hypercube Topology on Torus Multicomputers," IEEE trans. on Parallel and Distributed Systems, Vol.6, No.8, pp. 803-814, Feb 1995.



기우서

1986년 2월 순천대학교 영어교육과(학사). 2000년 2월 순천대학교 정보과학대학원(석사). 2003년~현재 순천대학교 컴퓨터과학과(박사과정). 관심분야는 컴퓨터교육, 정보보안, 알고리즘, 그래프 이론



김정섭

2006년 순천대학교 과학영재교육원 수료 (최우수상). 2009년 순천고등학교 졸업 2009년 현재 한국교원대학교 수학교육과 재학. 관심분야는 그래프이론, 알고리즘



이형욱

1994년 순천대학교 전산학과(학사). 1996년 전남대학교 전산통계학과(석사). 1999년 전남대학교 전산통계학과(박사). 1999년 10월~2002년 2월 한국정보사회진흥원(선임연구원). 2006년 1월~2007년 7월 University of Texas at Dallas(방문교수). 2002년 3월~현재 순천대학교 컴퓨터교육과 교수. 관심분야는 컴퓨터교육, 알고리즘, 그래프 이론



오재철

1978년 전북대학교 전기공학과(학사). 1982년 전북대학교 컴퓨터공학과(석사). 1988년 전북대학교 컴퓨터공학과(박사). 1986년~현재 순천대학교 컴퓨터과학과 교수 관심분야는 컴퓨터네트워크, 컴퓨터구조, 데이터통신, 분산네트워크