

# 유전자 알고리즘을 이용한 Passive Star 네트워크의 가상위상설계

정혜진<sup>†</sup>·위규범<sup>††</sup>·예홍진<sup>††</sup>·홍만표<sup>††</sup>

## 요약

과장분할 다중방식의 상호연결망의 결합구조의 설계 특징은 노드간에 실질적인 전송이 이루어지는 물리적인 위상과 물리적 링크가 가지는 여러 개의 채널을 노드의 송신장치와 수신장치에 할당한 가상적인 위상의 두 가지 측면을 고려할 수 있다. 교통량의 패턴에 따라 물리적인 위상의 변화 없이 가상위상만을 변경함으로써 상호연결망의 성능을 개선할 수 있으므로, 효율적인 가상위상의 설계는 매우 중요한 문제이다. 본 논문에서는 주어진 교통량의 평균 패킷전달 지연시간을 최소화하는 가상위상을 유전자 알고리즘을 이용하여 설계하는 방법을 제시한다.

## Virtual Topology Design of Passive Star Networks using Genetic Algorithms

Hea-Jin Jung<sup>†</sup> · Kyubum Wee<sup>††</sup> · Hong-Jin Yeh<sup>††</sup> · Manpyo Hong<sup>††</sup>

## ABSTRACT

We can consider the interconnection structure using WDM from two different levels, physical and virtual topologies. In the virtual topology, various channels on physical links can be established between transmitters and the receivers of the nodes. It is important to design efficient virtual topologies, because they have a benefit of performance improvement in interconnection networks depending on traffic matrices without changing physical topologies. In this paper we suggest a way to design virtual topologies that minimize average packet delays for given traffic matrices using genetic algorithms.

### 1. 서론

대규모 병렬(massively parallel) 시스템에서는 많은 노드들 사이에 대량의 데이터를 교환할 수 있어야 하므로 노드들이 이루는 결합구조의 성능은 대규모 병렬 시스템의 성능에 지대한 영향을 미친다. 최근에 광상

호결합구조(optical interconnection)의 기술 발달로 인하여 기존의 전자결합구조(electric connetion)의 한계를 극복할 수 있게 되었다. 특히 광결합구조에서 널리 쓰이는 데이터 전송방법의 하나가 광섬유의 거대 용량을 이용하는 과장분할 다중방식이다. 이것은 하나의 광섬유에 서로 다른 파장을 사용하는 여러 개의 독립적인 채널을 동시에 제공하여 데이터를 전송하는 방법이다. 이러한 초고속 통신 채널은 대규모 병렬 시스템에서 상호연결망(interconnection networks)을 구성하는데 효과적으로 사용될 수 있다[1, 2]. 하나의 예로서

\* 이 논문은 1998년도 정보통신부 대학기초 연구지원사업에 의해 작성되었음.

<sup>†</sup> 준회원 : 아주대학교 대학원 컴퓨터공학과

<sup>††</sup> 종신회원 : 아주대학교 정보통신전문대학원 교수

논문접수 : 1999년 3월 30일, 심사완료 : 2000년 1월 22일

passive star형 광상호연결망에서는  $N$ 개의 노드가 하나의  $N \times N$  passive star coupler를 통하여 서로 연결되어 있으며, 임의의 노드에서 정해진 파장(wavelength)을 사용하여 패킷을 전송하면 해당 노드에서 선택적으로 처리하는 방식으로 노드간의 패킷 전달이 이루어진다. 파장분할다중 방식을 이용하는 상호연결망은 보는 관점에 따라 노드간에 광케이블로 연결되어져 실질적인 전송이 이루어지는 물리적인 연결망과, 노드간에 물리적 링크가 가지는 여러 개의 파장을 노드의 송신장치와 수신장치에 할당된 가상적인 연결망으로 나누어 볼 수 있다. 다시 말해서 네트워크의 구성은 독립된 물리적인 위상(physical topology)과 가상적 위상(virtual topology)의 두 가지 형태로 표시될 수 있다. 그러므로 동일한 물리적인 연결에 수많은 가상적인 연결 형태가 구성될 수 있다. 이런 다양한 가상적인 연결 형태들 중에서 네트워크의 성능을 개선할 수 있는 적절한 가상적 연결을 물리적인 연결에 매핑한다. 이런 점에서 교통량 행렬(traffic matrix)의 형태에 많은 영향을 받는 네트워크에서 가상위상의 설계는 필수적이다. 이것은 주어진 교통량 행렬의 형태에 따른 노드간의 다양한 가상위상들 중에서 교통량 행렬의 형태에 가장 잘 부합되는 가상위상을 찾아내는 최적화 문제(optimization problem)가 된다. 최적화 문제 해결을 위한 많은 방법들이 연구되어 있으나, 본 논문에서는 최적화 문제에 좋은 성능을 나타내는 유전자 알고리즘을 적용하여 가상위상을 찾고자 한다.

본 논문의 구성은 다음과 같다. 제 2장은 해결하고자 하는 문제의 기술과 기존연구를 살펴보고, 제 3장은 제안한 유전자 알고리즘의 일반적인 개요에 대하여 설명한다. 제 4장은 유전자 알고리즘을 가상 위상 디자인에 적용한 방법을 설명하고, 제 5장은 유전자 알고리즘을 이용하여 설계된 가상위상을 여러 교통량 행렬의 형태에 적용한 시뮬레이션의 결과를 나타낸다. 마지막 장은 결론과 향후 연구과제를 논한다.

## 2. 문제 정의와 기존 연구

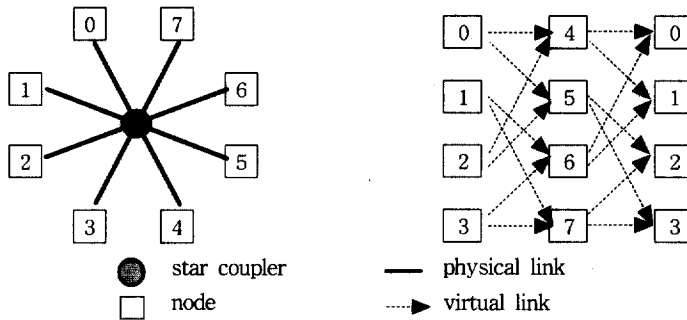
Passive Star형 상호연결망에서 가상위상의 설계는 하이퍼큐브(hypercube)나 셔플넷(shufflenet)과 같은 정형(regular)의 경우와 비정형(unregular)의 경우에 따라 차이를 가진다. 여기서, 정형이란 각 노드가 정해진 규칙에 의해서 다른 노드들과 연결되어 있는 경우이며,

비정형이란 임의의 패턴으로 다른 노드들과 연결되어 있는 경우를 말한다. 정형의 경우는 하이퍼큐브나 셔플넷과 같은 가상위상이 효율적이라는 것이 조사되어 있다[3]. 그러나 정형의 교통량 행렬의 형태에서도 노드 수에 따라 필요한 수만큼의 링크를 사용할 수 없는 경우나, 비정형의 교통량 행렬의 경우는 어떤 특정한 정형의 가상위상이 최적일 수 없다. 그러므로 이 논문에서는 정형의 가상위상을 그대로 적용할 수 없는 비정형의 교통량 행렬의 경우에 대한 가상위상의 설계를 주된 문제로 한다. 그리고 정형의 교통량 행렬의 형태에서 필요한 만큼의 링크를 가지지 못하는 가상위상의 설계도 고려한다.

각 노드는 고정된 파장만으로 송신과 수신 가능한 송신장치(transmitter)와 수신장치(receiver)를 하나 이상 이용한다. 단, 송신장치와 수신장치의 수는 같다. 논리적인 가상위상들 중에서 평균 패킷 전달 지연시간(delay time)을 최소화하는 목적함수(objective function)를 사용한다. 노드의 수는  $N$ , 각 노드가 가지는 각각의 송신장치와 수신장치의 수는  $T$ 로 표시한다. 여기서, 송신장치와 수신장치에 파장(wavelength)을 할당하는 것은 두 노드간 링크가 존재한다는 것을 의미한다. 그러므로 각 노드가 가지는 송신장치나 수신장치의 수는 결과적으로 두 노드간의 링크의 수가 된다. 각 노드는 하나의 파장으로 고정된 각각  $T$ 개의 송신장치와 수신장치를 가진다. 노드간의 링크 연결은 송신장치와 수신장치의 수 만큼인 차수(degree)를 가지는 그래프로 표시되며, 노드간의 링크를 하나 이상 가지는 멀티 그래프가 될 수도 있다. 그러므로 그래프는 간단히  $N \times T$ 의 행렬을 이용하여 하나의 가상위상(virtual topology)을 표현한다. (그림 1)은 8개의 노드가 Passive Star망으로 구성된 물리적 위상이 하나의 노드가 각각 두 개의 송신장치와 수신장치를 가진 가상의 Shufflenet을 구성할 수 있음을 보여준다.

이와 같이 Passive Star 네트워크를 이용하여 가상위상을 찾아내는 것은 매우 중요하다.

이런 가상위상 설계를 위한 기존의 연구로는 다음과 같은 방법들이 알려져 있다. 우선, 시스템의 개략화(approximation)로 네트워크를 모델링하는 방법과 이산적인(discrete) 최적화 문제로 파장 할당문제를 공식화하여 주어진 교통량 행렬에 대한 가상 위상을 구성하는 방법이 연구되었다[3,4]. 그러나 이런 방법들은 네트워크의 실질적인 동작을 충분히 반영하지 못했다.



(그림 1) 물리적 위상과 가상위상의 예

이에 반해 시뮬레이티드 어닐링(simulated annealing)[5, 6], 확률과정적 준거(stochastic ruler) 알고리즘[7], 휴리스틱(heuristic)한 방법은 주어지는 교통량 행렬에 대한 시뮬레이션을 통하여 구성하기 때문에 실질적인 네트워크의 동작을 반영하는 장점을 가진다. 그러나 긴 시뮬레이션 시간과 지역적인(local) 최적치에 머물러서 구하고자 하는 전역적인(global) 최적치를 얻어내기 어려운 경우가 있다. 본 논문에서는 유전자 알고리즘을 이용한 가상위상의 설계 방법을 제시한다.

### 3. 유전자 알고리즘(Genetic Algorithms)

유전자 알고리즘은 생명체의 적자생존과 진화현상에 기초한 확률적인 검색 알고리즘으로서, 주어진 조건에 잘 부합하는 답을 표현하는 문자열(string)로 인코딩(encoding)한 후보해(candidate)를 나타내는 개체들을 군집으로 구성한다. 이 군집에 교배, 돌연변이와 같은 유전 연산자의 반복적인 적용을 통하여 다음 세대에 더 좋은 해를 가지는 개체군으로 진화시킨다[8]. J. H. Holland에 의해 제안된 유전자 알고리즘 방법은 과학과 공학의 많은 분야에서 적절히 변형되어 사용되고 있다. 예를 들면 최적화 문제(optimization problem), 자동 프로그래밍(automatic programming), 생태학(ecology), 경제학(economics), 면역체계(immune systems), 유전학(population genetics), 사회체계(social systems)등의 분야에 응용되고 있다[8, 9]. 유전자 알고리즘의 개요는 다음과 같다.

- (0) 문제를 문자열로 인코딩한다.(문자열을 크로모솜이라 부른다.)
- (1) 크로모솜들을 무작위로 생성하여 초기 집단을 형성한다.
- (2) 집단의 각 개체들이 얼마나 좋은 후보해 인지를

나타내는 적합도를 계산한다. 여기서 적합도를 계산하는 규칙을 적합도 함수라고 부른다.

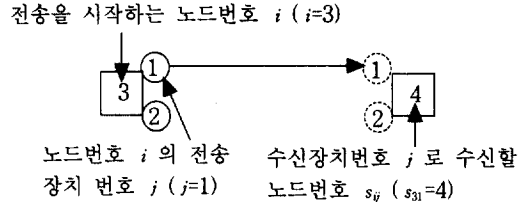
- (3) 두 개의 크로모솜을 선택하여 이들을 교배해서 얻은 두 개의 새로운 자손을 다음 세대의 군집에 더해준다. 여기서, 각 개체가 선택될 확률은 적합도가 높을수록 커진다. 이때 새로운 개체를 나타내는 크로모솜을 이루는 각 유전자는 돌연변이를 일으켜 그 값이 변할 수 있다. 이러한 과정을 반복하여 다음 세대의 군집을 형성한다.
- (4) (2), (3)과정을 정해진 회수만큼 또는 원하는 해가 생성될 때까지 반복한다.

유전자 알고리즘의 기본적인 개념은 적합도가 높은 개체가 가진 좋은 형질의 유전자들이 선택에 의하여 그 수가 불어나면, 교배를 통하여 서로 다른 개체에 있었던 좋은 형질의 유전자들이 같은 개체로 모이게 되며, 이에 따라 점점 적합도가 높은 후보해로 진화해나간다는 것이다. 여기서 돌연변이가 연산자의 역할은 크로모솜의 어떤 특정한 위치에 있는 유전자 값이 모든 개체에 걸쳐서 동일하게 되어 유전자의 다양성을 잃어버리는 경우가 발생하지 않도록 보장하는 장치이다. 주어진 문제를 유전자 알고리즘으로 해결하고자 하는 경우에 다음과 같은 점들을 설정해야 한다.

- (1) 문제의 후보해를 크로모솜인 스트링으로 인코딩하는 방법
- (2) 적합도 함수의 설계
- (3) 각 유전 연산자를 구현하는 방법
- (4) 각 유전 연산자를 적용하는 비율의 설정
- (5) 집단의 크기와 최대 세대수의 설정

위에서 나열한 점들을 어떻게 설정하느냐에 따라서 유전자 알고리즘의 성능은 많은 영향을 받는다.

Node \ Trans. No.	1	2
1	$s_{11}$	$s_{12}$
2	$s_{21}$	$s_{22}$
3	$s_{31}$	$s_{32}$
4	$s_{41}$	$s_{42}$



(그림 2) 개체를 구성하는 행렬의 형태

### 4. 유전자 알고리즘을 이용한 가상위상의 설계

#### 4.1 후보해의 인코딩(encoding)

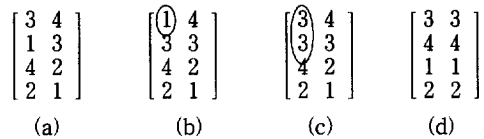
하나의 네트워크는  $N$ 개의 노드와 각 노드는  $T$ 개의 송신장치와 수신장치를 가지고 노드와 노드를 연결하는 임의의 파장이 할당된 가상위상(virtual topology)을 유향 그래프(directed graph)로 표시한다. 이런 유향 그래프가 하나의 개체를 나타낸다. 그래프의 표시를 간편하게 하기 위하여  $N \times T$ 의 행렬(matrix)을 이용하여 나타내며 노드와 송수신장치의 번호로 구성된 개체를 정의한다.

개체를 구성하는 열 번호는 각 노드가 가진 송신장치와 수신장치의 번호를 나타내며 행 번호는 네트워크가 가지는 노드의 번호를 나타낸다. 그리고 각 행렬이 가지는 원소(element)인  $s_{ij}$ 는  $i$ 번 노드가  $j$ 번 송신장치를 이용하여 송신한 패킷을  $j$ 번 수신장치로 수신할 수 있는 노드의 번호를 나타내는 것으로, 1부터  $N$ 사이의 임의의 정수 값을 가진다. 또한 두 노드간에 패킷의 전송은 반드시 송신장치의 번호와 수신장치의 번호가 같은 경우에만 가능한 것으로 간주한다. 예를 들면,  $s_{32}$ 는 노드 3번이 송신장치 2번을 통하여 패킷을 전송하고 2번 수신장치로 패킷을 수신할 수 있는 노드의 번호를 의미한다. 이런 원소의 값은 아래의 제약 조건을 만족하도록 구성한다.

- 1)  $s_{ii} \neq i, 1 \leq i \leq N, 1 \leq j \leq T$
- 2)  $s_{ij} \neq s_{kj}, 1 \leq i < k \leq N, 1 \leq j \leq T$

위에서 제시한 바와 같이 하나의 개체는 자기 자신의 노드로 전송되는 채널을 가지지 않아야 하며, 또한 서로 다른 노드들 간에도 같은 번호의 송신장치를 이용하게 되는 경우는 이미 채널을 할당받은 노드에 다시 채널을 설정할 수 없다. 즉,  $s_{11}$ 의 값이 3과 같으면

$s_{21}, s_{31}, s_{41}$ 의 값으로 노드번호 3번이 다시 선택될 수 없다. 이처럼 송수신장치의 번호별로 노드들의 값은 하나의 순열을 이루게 된다. 단, 노드 자신의 번호를 가지지 않는다. 따라서  $(s_{1j}, s_{2j}, s_{3j}, \dots, s_{kj})$ 은  $(1, 2, 3, \dots, k)$ 의 derangement가 된다. 또한, 개체를 임의로 생성하는 과정 중에서 비연결된(disconnected) 두 개 이상의 그래프로 나누어지는 개체의 생성은 배제되어야 한다. 이 조건은 송수신장치의 수가 많아질수록 비연결된 서브 그래프를 형성할 확률은 매우 작게 되나, 송수신장치의 수가 1인 경우는 유의해야 한다. 노드의 수  $N$ 이 4이고 송수신장치의 수  $T$ 가 2인 개체를 정의하면 다음과 같다. 노드 1번은 자기 노드 번호 1번을 제외한 2번부터 4번의 번호 중에서 무작위로 선택하여  $s_{11}$ 의 값을 결정한다. 노드 2번은 자기 노드 번호 2번과 노드 1번이 선택한 노드 번호를 제외한 2개의 노드 중에서 무작위로 하나를 선택하여,  $s_{21}$ 의 값을 결정한다. 이와 같은 방법으로 모두 8회 반복하여 송신장치에 대한 채널을 설정하면 (그림 3)의 (a)와 같은 가상위상을 나타내는 행렬을 얻어낼 수 있다.



(그림 3) 노드간의 링크를 나타내는 행렬들의 예

(그림 3)의 (b)는 원 부분인 1행 1열의 노드 값이 자기 자신의 노드 번호인 1이기 때문에 앞에서 제시한 제약 조건 1)에 만족되지 못하며, (c)는 제약 조건 2)를 만족하지 못한다. 마지막으로 (d)는 개체 정의를 위한 제약 조건을 만족하지만 비연결된(disconnected) 두 개의 그래프로 나누어지므로 잘못된 개체가 된다.

4.2 적합도 함수(Fitness Function)

주어진 교통량 행렬을 무작위로 구성한 하나의 가상 위상을 가진 유전자 알고리즘의 개체에 적용하여 각 패킷이 가지는 전달 지연시간을 계산한다. 전달 지연 시간은 패킷이 최초로 네트워크 상에 전송이 시작되는 시점에서 목적 노드에 도착하는 시점까지의 시간(time)으로 계산한다. 즉, 한 개의 패킷이 시간 2에서 전송을 시작하여 시간 6에서 목적한 노드에 도착하면, 이 패킷의 전달 지연시간은 4가 된다. 이렇게 계산된 패킷의 전달 지연시간의 합을 총 패킷의 수로 나누어 평균 전달 지연시간을 구한다. 아래의 수식은  $k$  번째 개체의 평균 패킷 전달 지연시간을 나타낸다.

$$D_k = \frac{\sum_{i=\alpha+1}^{\beta} d_i}{\beta - \alpha}$$

$d_i$ :  $k$  번째 개체가 나타내는 가상위상에서  $i$  번째 패킷이 목적 노드에 도달하는데 걸리는 전달지연시간

여기서  $\alpha$ ,  $\beta$ 는 주어지는 교통량 행렬이 가지는 패킷의 수에 따라 적절한 값으로 정의한다. 주어지는 교통량에 대해 초기와 후기의 패킷들이 교통량의 전달 지연시간에 큰 영향을 미치지 않기 때문에 모든 패킷을 고려하지 않는다. 예를 들면 교통량 행렬이 가지는 패킷의 수가 100이라면 그 중에서 초기의 패킷의 10개나 후기의 10개의 패킷은 교통량 전달지연시간에 큰 영향을 미치지 않기 때문에, 100개 패킷중에서 10번째부터 90번째의 패킷을  $\alpha$ ,  $\beta$ 의 값으로 설정하여 11번째 패킷부터 시작하여 패킷 전달지연시간을 결정한다. 그러므로  $d_{11}, d_{12}, \dots, d_i, \dots, d_{90}$  값의 합을 패킷수  $\beta - \alpha$  (=80) 값으로 나눈 값이 한 개체가 가지는 평균 패킷전달시간인  $D_k$ 가 된다. 개체의 수가  $n$ 일 때, 모든 개체들의 평균 패킷 전달 지연시간들( $D_1, \dots, D_k, \dots, D_n$ ) 중에서 최대값을  $Max$ 로 하여 각 개체가 가진 평균 패킷 전달 지연시간을  $\frac{Max - D_k}{Max}$  결과를  $Max$  값으로 나누어 적합도 값으로 한다. 그러므로 평균 패킷 전달시간이 작을수록 적합도 값이 높으며, 하나의 개체가 가지는 적합도 함수의 값  $S_k$ 는 0과 1사이의 실수의 값을 가지게 된다.

$$S_k = Fitness ( D_k ) = \frac{Max - D_k}{Max}$$

(  $S_k$  =  $k$  번째 개체의 적합도 값 )

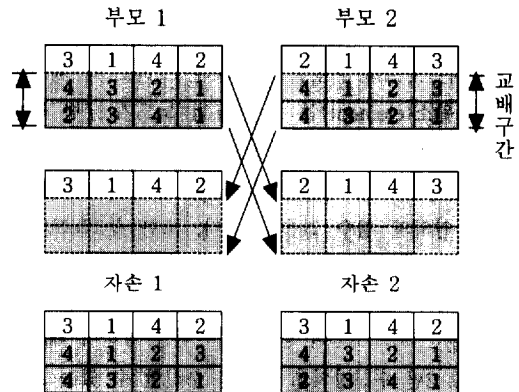
4.3 선택(Selection)

교배나 돌연변이 연산을 적용할 개체를 선택하는 방법으로, 널리 사용되고 있는 룰렛 휠(roulette wheel) 선택을 사용하였다. 이때 한 군집 내에서 적합도 값이 가장 좋은 개체를 교배나 돌연변이 연산을 적용하지 않은 상태로 다음 세대에 그대로 복사함으로써 좋은 유전자를 다음 세대에 존속할 수 있도록 하는 정예주의(elitism) 선택을 가미하였다.

4.4 유전 연산자(Genetic Operators)

4.4.1 교배(Crossover)

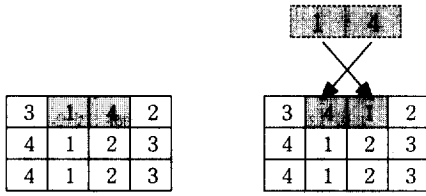
두 개의 부모 개체에서 송신장치 번호를 가지고 교배할 지점을 임의로 선택하여 두 개의 개체의 일부분의 원소들을 서로 교환한다. (그림 4 참조)



(그림 4) 교배연산

4.4.2 돌연변이(Mutation)

교배나 복제 연산이 이루어진 후에 새로 생성된 개체에서 유전자의 값을 바꾸는 것으로, 임의의 송신장치의 번호와 돌연변이가 직접 이루어질 두 개의 지점을 선택하여 서로의 값을 교환하는 상호교환(reciprocal exchange)연산을 이용한다 (그림 5 참조).



(그림 5) 돌연변이 연산

4.5 제어변수(Control Parameters)

군집의 크기(population size)는 30, 교배비율(cross-over rate)은 0.75, 돌연변이 비율(mutation rate)은 0.01, 정예주의(elitism) 비율은 10%로 하였으며, 세대수는 교통량 행렬의 형태에 따라 정형의 경우는 150, 비정형인 경우는 70으로 달리 설정하였다. 이것은 특정한 노드간에만 전송할 패킷이 발생하는 정형의 교통량 행렬에서는 넓은 탐색공간에서 최적화된 가상위상들이 차지하는 비율이 절대적으로 작다. 반면에 자기 자신을 제외한 모든 노드간에 전송할 패킷을 가지는 비정형의 경우는 탐색공간 내에서 적합도가 높은 가상위상들이 차지하는 비율이 정형의 경우보다 상대적으로 크기 때문에 적은 세대수에서도 좋은 가상위상을 찾을 수 있기 때문이다.

5. 실험 및 결과

실험에서 사용하는 모든 교통량 행렬은 각 노드가 자신에게 패킷을 보내지 않으므로 교통량 행렬의 대각선은 모두 0이다. 그 외 다른 모든 엔트리들의 값은 패킷 도달 비율(packet arrival rate)  $\lambda$  값에 의하여 각 노드가 전송할 패킷의 수를 결정하게 된다. 여기서의 패킷의 수는 포아송(Poisson) 분포에 따른다. 교통량 행렬은 (1) 노드간의 패킷 도달 비율( $\lambda$ )을 일정하게 한 경우, (2) 노드간의 패킷 도달 비율( $\lambda$ )을 다양하게 한 경우, (3) 준균일한 경우, (4) 정형의 경우로 분류하여 가상위상을 실험한다. 한 노드에서 다른 노드로 패킷을 전송하기 위한 시간은 unit time 값 2로 한다. 평균 패킷 전달 지연시간을 구하기 위해 교통량 행렬이 가지는 모든 패킷을 고려하지 않고 시간의 흐름상 중간의 패킷을 고려한다. 즉, 비정형의 교통량 행렬이 가지는 모든 패킷에서 각 노드가 가지는 송신장치를 통하여 목적노드에 단일 흐름에 보낼 수 있는 패킷의 수만

큼을 앞부분과 뒷부분에서 제외한다. 이 구간을 4.2절에서와 같이  $[\alpha, \beta]$ 로 나타내며, 이 구간의 평균 패킷 전달 지연시간을 구한다. 교통량 행렬이 가지는 패킷의 수는 노드의 수와 패킷 도달 비율  $\lambda$ 에 의해 결정된다. 예를 들어, 노드의 수(N)가 16이고  $\lambda$  값이 2인 경우 16개의 노드가 자기 자신을 제외한 15개의 노드에 전송할 패킷이 존재한다고 할 경우 패킷의 수는 노드수  $\times$  (노드수 - 1)  $\times \lambda$  인  $16 \times 15 \times 2 = 480$  개 정도가 됨을 볼 수 있다. 또한, 송수신장치의 수(T)가 3인 경우에 16개의 노드가 한번에 보낼 수 있는 패킷의 수는 노드수  $\times$  송수신장치수  $\times \lambda$  인  $16 \times 3 \times 2 = 96$  개로 계산된다. 그러므로 평균 패킷 전달지연시간 계산 구간에서 처음 96개의 패킷과 마지막 96개의 패킷을 제외한다. 노드의 수와 송수신장치 수의 증가에 따라 구간의 값은 일정하게 적용한다. 시뮬레이션의 결과표에서 사용되는 기호는 다음과 같다.

**Number of Sim.** : 주어진 교통량 행렬에 대하여 유전자 알고리즘을 적용하여 가상위상을 설계하는 작업의 반복횟수.

**Init.** : 초기에 구성된 가상위상에 교통량 행렬이 적용된 경우 얻어지는 평균 패킷 전달 지연시간(average packet delay).

**Avg.** : 주어진 교통량 행렬에 대하여 Number of Sim. 번 유전자 알고리즘을 적용하여 얻어낸 가상위상들(virtual topology)의 평균 패킷 전달 지연시간의 평균.

**Best Sol.** : 하나의 교통량 행렬에 대하여 Number of Sim. 번 시뮬레이션한 후 얻어진 가장 좋은 가상위상의 평균 패킷 전달 지연시간.

5.1 각 노드간의 패킷 도달 비율( $\lambda$ )을 2로 고정한

교통량 행렬

노드 수가 16인 경우는 구간을 [100, 400]으로 하였으며, 노드 수가 32인 경우 [200, 1800]으로 하여 평균 패킷 전달 지연시간을 계산할 패킷을 설정한다. 노드 수가 클수록 수행시간이 길기 때문에 노드 수가 16인 경우 시뮬레이션 횟수는 50, 노드 수가 32인 경우 20으로 달리 적용하여 수행시간을 유사하게 하였다. 노드의 수가 16이고 송수신장치의 수가 3인 경우 제안된 알고리즘에 의해 얻어진 가상위상이 초기에 임의로 구성한 가상위상에서 얻어진 평균 패킷 전달 지연시간보다 약 39%의 개선을 가져왔다. <표 1>은 평균 패킷 전달 지연시간의 결과이다.

<표 1>  $\lambda = 2$ 인 평균 패킷 전달 지연시간

N	T	Init.	Avg.	Best Sol.	(1-Avg./Init.)%
16	3	22.60	13.82	12.65	39
	4	14.52	10.29	9.58	29
	5	10.90	8.16	7.29	25
32	3	75.42	47.91	40.61	36
	4	58.75	37.76	35.88	36
	5	42.53	28.96	27.67	32

노드의 수가 16인 경우에는 송수신장치의 수가 4인 경우에 하이퍼큐브와 서플넷인 정형의 가상위상을 적용할 수 있다. 이렇게 정형의 가상위상을 적용하여 얻어진 패킷 전달지연시간은 하이퍼큐브 32.12, 서플넷은 42.45의 값을 얻었다. 이 결과는 정형의 가상위상을 적용할 조건을 만족하더라도 비정형의 교통량 행렬이 주어지는 경우는 정형의 가상위상보다 제안된 가상위상 설계 알고리즘을 이용하여 얻어진 가상위상이 더 효율적임을 알 수 있다.

5.2 노드에 따라 패킷 도달 비율( $\lambda$ )이 달리 적용된 교통량 행렬

노드간의  $\lambda$ 값을 1과 10사이에서 다양하게 선택하여 교통량 행렬을 구성한다. 즉 전송과 수신할 두 노드간의 패킷의 수가 일정한  $\lambda$ 값을 가지지 않고 노드간에서 다른  $\lambda$ 값을 가지는 경우이다. 예를 들어 노드 1에서 노드 2에 전송되는  $\lambda$ 값이 2, 노드 3에서 노드 5에 간에는  $\lambda$ 값이 7인 형태인 경우이다. ( $\lambda$ 값이 클수록 패킷의 수는 많아지기 때문에 노드간 전송될 패킷의 수가 큰 차이를 가진다.) 이것은 (그림 6) (a)의 교통량 행렬에서 노드 5번이 노드 0번으로 전송할 패킷의 수가 과도하게 늘어나 과열점(hot spot)이 생성되는 것으로부터 알 수 있다.

	0	1	2	3	4	5	6	7
0	0	3	6	4	6	8	2	3
1	3	0	7	9	1	6	1	7
2	7	1	0	9	9	3	6	7
3	7	1	3	0	9	1	1	7
4	1	7	1	3	0	9	2	1
5	10	4	9	1	8	0	9	1
6	2	8	2	1	6	9	0	1
7	1	9	2	1	2	9	8	0

(그림 6) N = 8, T = 3인 경우의  $\lambda$  값이 상이한 교통량 행렬

노드 수가 16이고 각 노드가 가지는 송신장치와 수신장치의 수가 3인 경우 평균 패킷 전달 지연시간의 결과는 <표 2>와 <표 3>에 나타난다. 여기서 <표 1>의 경우는  $\lambda$ 의 값을 두가지로 고정된 경우이고 <표 2>의 경우는  $\lambda$ 의 값을 두 구간내의 임의의 값을 가지도록 구성하여 교통량 행렬의 패킷의 형태가 다른 경우를 나타낸다. <표 1>의 결과에서  $\lambda$ 값이 1과 10일 때 초기의 평균 패킷 전달 지연시간에 비해 제안된 알고리즘이 적용된 후 얻어지는 가상위상이 44%의 개선 정도를 나타낸다.  $\lambda$ 값의 차이가 작은 1과 4인 경우는 34%로 1과 10인 경우에 약 1.3배정도 더 개선되었다.

<표 2> 두 개의  $\lambda$  값을 적용한 경우의 평균 패킷 전달 지연시간

$\lambda$ 값	Init.	Avg.	Best Sol.	(1-Avg./Init.)%
1, 10	32.17	18.03	16.15	44
1, 7	27.07	17.37	15.56	36
1, 4	21.82	14.42	12.80	34
1	15.12	9.97	9.47	34

<표 3>과 같이 주어진 구간에서 임의의  $\lambda$ 을 적용하여 생성한 교통량 행렬에 대해서는 구간의 값이 큰 경우가 작은 경우보다 약 1.6배의 개선 정도를 얻었다.

<표 3> 주어진 구간에서 임의의  $\lambda$  값을 적용한 경우의 평균 패킷 전달 지연시간

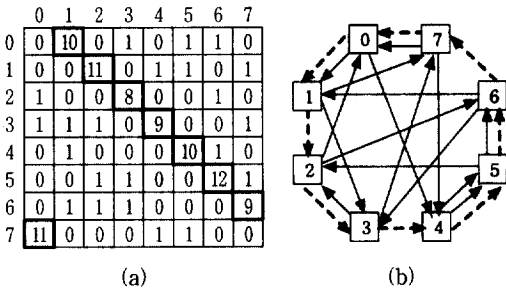
$\lambda$ 값	Init.	Avg.	Best Sol.	(1-Avg./Init.)%
1~10	32.24	18.29	15.69	51
1~7	29.77	17.35	14.85	42
1~4	21.07	14.24	13.04	32
1	15.12	10.33	9.43	32

<표 2>, <표 3>의 결과로부터 모든 노드간의 패킷 도달 비율( $\lambda$ )의 값에 차이가 클수록 제안된 알고리즘이 좀 더 효율적임을 알 수 있다. 송수신장치의 수가 3이 아닌 4나 5인 경우에도 마찬가지로의 결과를 얻어낼 수 있으며, 노드 수를 32개로 확장하여도 유사한 결과를 얻어내리라고 예상된다.

5.3 준균일(quasiuniform) 교통량 행렬

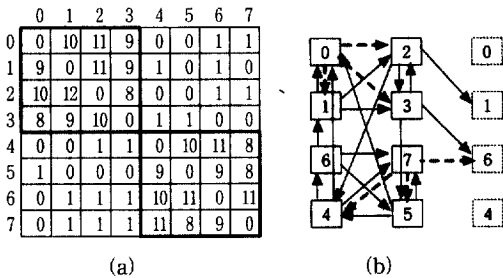
준균일한 교통량 행렬이란 정형의 가상위상을 따르는 정형의 교통량 행렬의 변형이다. 즉 정형의 링(ring)은 노드 0 → 노드 1 → 노드 2 → 노드 3 →

노드 4 → 노드 5 → 노드 6 → 노드 7 → 노드 0 사이에 방향성을 가지는 노드간에만 전송할 패킷을 가지는 것이나 준균일 교통량 행렬은 앞에 열거한 노드간 이외의 노드간에도 전송되어질 소수의 패킷을 가진다. 단지, 링 형태의 노드간에 과열점(hot spot)을 가진다. (그림 7) (a)의 교통량 행렬에 대한 초기 가상위상의 평균 패킷 전달 지연시간은 9.43이다. 그러나 유전자 알고리즘을 적용한 후 얻어낸 가상위상은 3.52로 약 63%의 감소를 가져왔다. (그림 7) (b)인 가상위상의 결과에서는 링 형태의 과열점이 존재하는 노드간에 한 홉에 패킷의 전송이 가능하도록 링크가 구성되었으며, 과열점을 해결하기 위해 노드간의 링크가 대부분 두 개의 링크를 가지는 것을 볼 수 있다. 예를 들어 노드 0과 노드 1사이, 노드 7과 노드 0, 노드 4와 노드 5등에 링크가 2개 존재한다.



(그림 7) N = 8, T = 3인 유사 링(Quasi Ring) 형태

- (a) 교통량 행렬
- (b) 유전자 알고리즘으로 구한 가상위상



(그림 8) N = 8, T = 3인 유사 비연결(Quasi disconnected) 형태

- (a) 교통량 행렬
- (b) 유전자 알고리즘으로 구한 가상위상

(그림 8)은 비연결된(disconnected) 교통량 행렬과 패킷의 전송을 위한 링크의 구성을 나타낸다. 이 교통량 행렬의 형태는 노드들을 두 그룹으로 나누어 각 그룹내의 노드간에 과열점이 발생한다. 제안된 알고리즘이 찾아낸 가상위상도 두 그룹 내에서 노드간의 링크가 밀집된 것을 볼 수 있다.

정형의 가상위상 구조를 따르는 교통량 행렬의 특수한 형태에 대하여 제안된 알고리즘이 효율적으로 적용되는 것을 알 수 있다. <표 4>은 준균일 교통량 행렬에 대한 결과를 나타낸다.

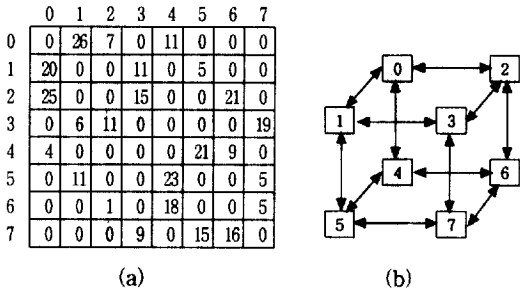
<표 4> 준균일 교통량 행렬에 따른 평균 패킷 전달 지연시간

Traffic Pattern	Init.	Avg.	Best Sol	(1-Avg./Init.) %
quasi ring	9.43	3.52	3.11	63
quasi disconnected	9.37	3.43	2.90	63

5.4 정형의 교통량 행렬

정형의 교통량 행렬은 네트워크를 구성하는 노드 수에 따라 특정한 노드간에만 전송할 패킷을 가진다. 예를 들어, 하이퍼큐브 형태를 가진 교통량 행렬에 노드 수가 8이면 각 노드는 패킷을 전송할 노드 수가 3이 된다. 정형의 교통량 행렬에서 한 노드가 패킷을 보낼 노드 수만큼의 송수신장치를 가지는 경우는 정형의 가상위상이 최적이 된다. 그러나 패킷을 보낼 노드 수만큼의 송수신장치를 가지지 못하는 경우에는 정형의 가상위상이 항상 최적일 수 없다. 노드의 수가 8인 하이퍼큐브 형태를 가지는 정형의 교통량 행렬에 각 노드의 송신장치와 수신장치의 수를 3으로 하여 제안된 알고리즘을 적용한 결과는 초기의 평균 패킷 전달 지연시간이 7.86에서 평균 패킷 전달 지연시간이 2.15인 가상위상을 얻었다. (그림 9)에서 보듯이 정확히 하이퍼큐브 형태의 가상위상을 얻었다. (그림 10)에는 서플넷 형태의 교통량 행렬에 대한 결과로 유전자 알고리즘에서 초기에 생성되는 개체들의 평균 패킷 전달 지연시간은 11.82에서 3.09로 약 74%의 개선정도를 얻었으며 최적의 가상위상인 서플넷을 찾았다. 위의 두 경우의 결과에서 보듯이 제안된 알고리즘에 의해 찾아지는 가상위상이 타당하다는 것을 보여주는 것이다.





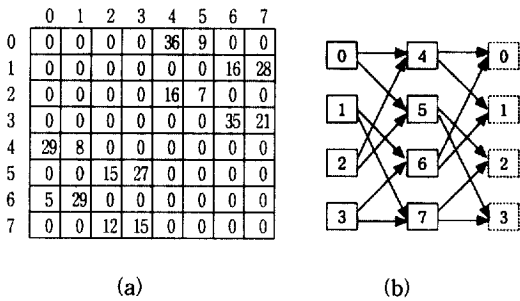
(a)

(b)

(그림 9) N = 8, T = 3인 하이퍼큐브

(a) 교통량 행렬

(b) 유전자 알고리즘으로 구한 가상위상



(a)

(b)

(그림 10) N = 8, T = 2인 셔플넷

(a) 교통량 행렬

(b) 유전자 알고리즘으로 구한 가상위상

<표 5>는 노드 수를 16으로 확장하고 각 노드가 가지는 송신장치와 수신장치의 수를 2, 3, 4, 5로 하여 시뮬레이션 된 결과로 대략 49%정도의 전달 지연시간의 감소를 나타내고 있다.

<표 5> N = 16인 평균 패킷 전달 지연시간

Traffic Pattern	T	Init.	Avg.	Best Sol.	(1-Avg./Init.) %
hyper cube	2	30.60	14.67	12.7	52
	3	18.49	9.57	8.95	48
	4	17.07	8.12	7.12	52
	5	12.23	7.64	6.28	38
shuffle net	2	30.43	15.20	12.80	50
	3	19.32	10.03	8.24	47
	4	15.455	7.79	6.86	50
	5	12.355	7.57	6.76	39

노드 수가 16개이고 송수신장치가 4개인 경우는 노드가 8개인 경우에 반해 정형의 하이퍼큐브나 셔플넷의 가상위상을 얻어내지는 못하였다. 이는 임의로 구성되어지는 가상위상이 정형의 하이퍼큐브 구조를 만족하는 노드를 선택할 확률이 매우 작기 때문이다. 다시 말해서 16개 노드로 구성된 가상 링크의 매핑방법은 약 15!개이다. 그러나 하이퍼큐브 구조를 이루기 위해서는 15개의 노드 중 이미 설정되어진 4개의 노드를 선택하는 문제로 매핑의 방법이 축소되기 때문에 15!개의 매핑방법을 전제로 개체를 생성하는 유전자 알고리즘으로는 정확한 하이퍼큐브의 가상위상을 얻어내는 것은 쉽지 않다. 그러나 논문에서 실행한 개체의 수나 세대수를 좀 더 늘려 시뮬레이션이 된다면 노드 수가 8인 경우에서 얻는 결과와 같이 하이퍼큐브나 셔플넷 구조의 가상위상을 찾아낼 수 있으리라 보인다.

각 노드가 가지는 송신장치와 수신장치의 수가 필요보다 작은 2나 3인 경우와 필요보다 많은 5인 경우와의 결과에서 보면 초기의 평균 패킷 전달 지연시간과 제안된 알고리즘이 적용된 후 찾아진 가상위상의 평균 패킷 전달 지연시간은 모두 약 40%정도의 개선을 가져왔다. 하이퍼큐브의 교통량 행렬에 송수신장치의 수가 4개인 하이퍼큐브 가상위상을 그대로 적용하여 얻어지는 평균 패킷 전달 지연시간은 6.36이다. 이 경우에 유전자 알고리즘을 적용하면 <표 5>에서 보듯이 평균 패킷 전달지연시간이 최적치 6.3에 가까운 7.12이다. 이러한 점으로 미루어 볼 때 송수신장치의 수가 2, 3, 또는 5개인 경우에도 유전자 알고리즘이 최적에 근접한 가상위상을 찾아내고 있다고 생각된다.

5.5 확률과정적 준거(Stochastic Ruler) 알고리즘과

제한된 알고리즘의 차이점

기존의 확률과정적 준거 알고리즘에 의한 가상위상의 설계[7]와 본 논문에서 제안한 유전자 알고리즘을 이용하는 방법을 비교해 본다. 가상위상 설계를 위한 시뮬레이션의 방법들은 임의로 구성된 가상위상을 사용하여, 교통량 행렬에 의해 실질적으로 패킷을 노드 간에 전달하여 가장 좋은 목적함수 값을 가지는 가상위상을 얻어내는 것이다. 확률과정적 준거 알고리즘은 다음과 같은 단계로 수행된다.

- (1) 임의의 초기 가상위상을 구성한다. (유전자 알고리즘의 개체 생성과 유사함)

- (2) 초기 가상위상에서 한 노드가 이웃하는 노드와 링크를 바꾸어서 생성되는 새로운 가상위상을 이웃 상태(state)로 정의한다.
- (3) 이웃 상태와 현재의 상태를 상대적인 비교를 통하여 두 상태 중에서 이웃 상태가 좋은 목적함수 값을 가지면 현 상태의 가상위상을 이웃 상태로 변화하고 그렇지 않으면 현 상태에 머물러 있게된다.
- (4) (2), (3)를 반복하여 실행하다가 하나의 상태에서 일정한 반복 횟수동안 변화가 없으면 실행이 종료된다.

초기의 가상위상에서 이웃하는 가상위상들을 탐색의 대상으로 하기 때문에 넓은 탐색공간 내에 존재하는 다양성을 제공하지 못하며, 최적치에 근사한 값을 얻어내기 위한 긴 수행시간(많은 반복횟수)을 필요로 한다. 반면에 제안된 알고리즘은 탐색공간 내에 존재하는 다양한 가상위상을 탐색의 대상으로 하며 수행시간을 단축할 수 있다.

<표 6> 비정형의 교통량 행렬에 대한 평균 패킷 전달 지연시간

T	Init.	Algorithms applied	Avg	(1-Avg./Init.) %
3	18.49	GA	10.29	44
		SR	11.97	35
4	17.07	GA	7.45	56
		SR	9.72	43
5	12.23	GA	6.89	44
		SR	8.98	27

<표 6>의 결과는 노드 수 16개로 두 개의 알고리즘의 반복횟수를 비슷하게 하여 시뮬레이션한 결과이다. 즉, 확률과정적 준거 알고리즘의 반복횟수는 1800~2200이었으며, 유전자 알고리즘에서는 군집의 크기 30과 세대수 70으로 반복횟수가 2100 정도로 하였다. 여기서 GA는 제안된 유전자 알고리즘이고, SR은 확률과정적 준거 알고리즘을 나타낸다. 비정형의 교통량 행렬이 적용된 경우는 유전자 알고리즘이 확률과정적 준거 알고리즘보다 평균 약 1.4배정도 더 개선된 것을 볼 수 있다.

정형의 경우에는 노드 수 16개로 반복횟수가 약 6000번으로 송수신장치의 수를 달리하여 얻은 결과가 <표 7>이다. 비정형의 경우와 같이 약 1.3배의 평균

<표 7> 정형의 교통량 행렬에 대한 평균 패킷 전달 지연시간

T	Init.	Algorithms applied	Avg.	(1-Avg./Init.) %
3	22.60	GA	13.82	39
		SR	14.25	31
4	14.52	GA	10.29	29
		SR	11.14	23
5	10.90	GA	8.16	25
		SR	8.85	19

패킷 전달 지연시간이 감소한 것을 볼 수 있다. 이런 결과로부터 네트워크의 노드 수와 송수신장치의 수가 많아졌을 때 기존의 확률과정적 준거 알고리즘에 비해 제안된 유전자 알고리즘이 더욱 효율적이라 예상된다.

## 6. 결 론

파장분할다중방식을 이용한 Passive Star형 네트워크에서 송수신장치가 고정된 파장을 가지는 경우에 교통량 행렬에 따른 가상위상(virtual topology) 설계를 위한 효율적인 방법을 제안하였다. 제안된 유전자 알고리즘을 이용하여 비정형의 교통량 패턴과 정형의 교통량 패턴에 대해서 실험하였다. 포아송(poisson) 분포의  $\lambda$  값에 따른 비정형 교통량 행렬은 노드 수에 관계없이 각 노드가 가지는 송수신장치의 수가 많을수록 평균 패킷 전달 지연시간이 감소되는 가상위상을 얻었으며, 초기에 임의로 구성되는 가상위상보다 현저한 개선정도를 가져왔다. 비정형의 교통량 행렬에 대하여도 하이퍼큐브(hypercube)나 셔플넷(shuffle net)과 같은 정형의 가상위상을 사용하는 것보다 효율적임을 볼 수 있었다. 확률과정적 준거 알고리즘과의 비교에 있어서는 정형의 교통량 행렬의 경우가 비정형의 경우보다 더 높은 평균 패킷 전달 지연시간의 개선정도를 보였다. 이처럼 제안된 알고리즘은 passive star망들로 이루어진 넓은 탐색공간에서 교통량 행렬의 패턴에 따른 적절한 가상위상을 찾아내었다. 앞에서 설명한 바와 같이 가상위상의 설계에 유전자 알고리즘이 좋은 성능을 보였으므로, 노드의 수와 송수신장치의 수가 많을수록 또는 파장할당(wavelength assignment), 파장 라우팅(wavelength routing)을 하는 경우에는 가상위상의 탐색공간이 더 커지며, 이 경우에는 기존의 시뮬레이티드 어닐링(simulated annealing)이나 확률과정

적 준거(stochastic ruler) 알고리즘 보다 유전자 알고리즘(genetic algorithms)이 효과적인 결과를 얻을 것으로 기대된다.

향후 연구 과제로는 우선 본 논문의 passive star형의 네트워크로 물리적인 상호연결이 star coupler에 의해 모든 노드간의 링크를 가지는 것이 아니라 노드간의 링크가 제약 가지는 물리적 상호연결망의 경우에도 적용 가능하도록 일반화해 보는 것이다. 또한, 모든 노드가 고정된 파장을 이용하는 같은 수의 송수신장치를 가지는 것이 아니라 여러 파장으로 조율이 가능한 송수신장치를 가지고 채널을 공유하게 되는 경우로 확대하여 가상위상을 설계하는 문제에 유전자 알고리즘을 적용하는 것도 흥미 있는 연구과제라고 생각된다.

### 참 고 문 헌

- [1] C. A. Brackett, "Dense Wavelength Division Multiplexing Networks : Principles and Applications," IEEE J. on Selected Areas in Communications. Vol. 8, No.6, pp.948-964, Aug. 1990.
- [2] R. J. Vetter and D. H. C. Du, "Distributed Computing with High-Speed Optical Networks," IEEE Computer, pp. 8-18, Feb. 1993.
- [3] J. A. Bannister, L.Fratta, and M. Gerla, "Topological Design of the Wavelength Division Optical Networks," IEEE INFOCOM '90, San Francisco, CA, June. 1990.
- [4] J. F. P. Labourdette, and A. S. Acampora, "Logically Rearrangeable Multihop Lightwave Networks," IEEE Trans. Communication, Vol.39, Aug. 1991.
- [5] S. W. Bollinger and S. F. Midkiff, "Heuristic Technique for Processing and Link Assignment in Multicomputers," IEEE Trans. Comput., Vol.40, Mar. 1991.
- [6] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by Simulated Annealing," Science, Vol.220, No.4598, May 1983.
- [7] A. Ganz, and X. Wang, "Efficient Algorithm for Virtual Topology Design in Multihop Lightwave Networks," IEEE/ACM Trans. on Networking, Vol. 2, No.3, June. 1994.
- [8] D. E. Goldberg, "Genetic Algorithms in Search, Optimization, and Machine Learning," Addison-Wesley, 1989.
- [9] M. Mitchell, "An Introduction to Genetic Algorithms," MIT Press, 1996.



### 정 혜 진

e-mail : jhjsbg@madang.ajou.ac.kr  
 1993년 경기대학교 전자계산학과 (학사)  
 1999년 아주대학교 컴퓨터공학과 (석사)  
 2000년~현재 원곡고등학교 교사

관심분야 : 컴퓨터 이론, 유전자 알고리즘



### 위 규 범

e-mail : kbwee@madang.ajou.ac.kr  
 1978년 서울대학교 수학과(학사)  
 1984년 University of Wisconsin 전산학과(석사)  
 1992년 Indiana University 전산학과(박사)

1993년~현재 아주대학교 정보 및 컴퓨터공학부 교수  
 관심분야 : 컴퓨터 이론

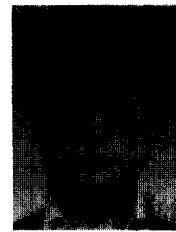


### 예 흥 진

e-mail : hjyeh@madang.ajou.ac.kr  
 1986년 서울대학교 수학교육과(학사)  
 1988년 아주대학교 전자계산학과 (석사)  
 1990년 Univ. Joseph Fourier-INP Grenoble(佛), 응용수학과 (D.E.A.)

1993년 Univ. Claude Bernard - ENS Lyon(佛), 전자계산학과(박사)

1993년~현재 아주대학교 정보 및 컴퓨터공학부 교수  
 관심분야 : 컴퓨터 산술, 병렬 알고리즘과 구조, VLSI 알고리즘과 구조



### 홍 만 표

e-mail : mphong@madang.ajou.ac.kr  
 1981년 서울대학교 계산통계학과 (학사)  
 1983년 서울대학교 계산통계학과 (석사)  
 1991년 서울대학교 전산과학과(박사)

1985년~현재 아주대학교 정보 및 컴퓨터공학부 교수  
 관심분야 : 병렬처리, 광상호연결망, 시스템 성능분석