

# 최소가중치삼각화 문제를 위한 개선된 유전자 알고리즘

이범주<sup>1</sup> · 한치근<sup>2</sup>

<sup>1</sup>FID 기술연구소 / <sup>2</sup>경희대학교 전자정보학부

## A Modified Genetic Algorithm for Minimum Weight Triangulation

Bum-Joo Lee<sup>1</sup> · Chi-Geun Han<sup>2</sup>

The triangulation problem is to make triangles using the given points on the space. The Minimum Weight Triangulation(MWT) is the problem of finding a set of triangles with the minimum weight among possible set of the triangles. In this paper, a modified genetic algorithm(GA) based on an existing genetic algorithm and multispace smoothing technique is proposed. Through the computational results, we can find the tendency that the proposed GA finds good solutions though it needs longer time than the existing GA does as the problem size increases.

### 1. 서론

삼각화는 2차원 평면상에 주어진  $n$ 개의 점들을 서로 교차하지 않게 연결한 선분들을 연결하여 삼각형들의 집합을 구성하는 문제이다. 하나의 점 집합에 대하여 삼각화 방법은 많은 경우가 있으며, 그 중 최소의 선분 길이 합을 갖는 삼각화를 찾는 문제를 최소가중치삼각화(Minimum Weight Triangulation: MWT) 문제라 하며, 이러한 MWT 문제는 NP-hard로 알려져 있다 (Garey and Johnson, 1979; Preparata and Shamos, 1985).

현재 일반적으로 사용되는 MWT 문제 해결 방법으로는 그리디 삼각화 방법(Greedy Triangulation Method)과 딜러니 삼각화가 있는데, 이들 방법은 최적해에 접근하지 못한다는 것이 증명되어 있다(Kirkpatrick, 1995; Manacher and Zobrist, 1979). 최근, Qin, Wang, Gong (1997)은 MWT의 최적해를 구하고자 유전자 알고리즘을 적용하여 문제를 해결하였는데 본 논문에서는 유전자 알고리즘을 여러 번 수행하는 다단계 탐색공간 스무딩 기법을 이용한 효율적인 문제 해결 방법을 제시하고 실험을 통하여 이의 효용성을 밝힌다. Qin, Wang, Gong (1997)의 기존 연구에서는 기본적인 유전자 알고리즘의 방법을 제시하고 있다.

MWT 문제에 유전자 알고리즘을 적용하여 해를 구할 때, 주어진 문제의 점의 분포를 일정한 방법으로 조정하여 선분의 길이를 되도록 균일하게 만들어 해를 구하는 방법을 생각할 수 있다. 이 방법은 다단계 탐색공간 스무딩 방법으로 Gu와 Huang (1994)에 의하여 처음 제안되었다. 이는 원래 문제 해 공간의 패턴을 지닌 여러 개의 스무딩 공간을 생성하게 되는데,

가장 완전한 탐색 공간으로부터 시작하여 각 단계에서 얻어지는 해 집합을 다음 단계의 초기해로 설정하여 해를 구하는 과정을 반복하는 방법이다. 주어진 점 집합의 전체적인 패턴을 살펴보면, 단계적으로 연결 형태를 고려하면서 점차적으로 세부적인 연결 형태를 결정한다면 보다 좋은 결과값을 얻을 수 있을 것이다. 본 논문에서 적용한 스무딩 방법은 문제의 점 각각의 위치를 하나의 점을 기준으로 하여 일정한 거리가 되도록 이동시키고, 점차적으로 원래의 위치로 복원시키는 과정을 진행하여 여러 개의 탐색공간을 생성하였다.

2장에서는 MWT 문제를 설명하고, 3장에서는 MWT 문제에 적용된 유전자 알고리즘을 설명한다. 4장에서는 탐색공간 스무딩을 적용하기 위한 MWT 문제의 변형을 설명하고, 5장에서는 실험 및 성능평가를 설명한 후 6장에서 결론을 맺는다.

### 2. Minimum Weight Triangulation

2차원 평면공간 상에  $n$ 개의 임의의 점 집합  $C$ 가 주어졌을 때, 점 집합  $C$ 의 점들을 기존에 연결된 선분과 교차하지 않도록 새로운 선분으로 연결하여, 더 이상 연결할 점이 존재하지 않을 때 얻어진 평면 분할을 삼각화(Triangulation)라 한다(Preparata and Shamos, 1985).

<그림 1>은 평면상의 5개의 점이 주어졌을 때 구할 수 있는 삼각화의 2가지 예를 보인 것이다. 이 예에서와 같이 평면상에 주어진 하나의 점 집합에 대하여 다수의 삼각화를 만들 수 있다. MWT 문제는 다음과 같이 정의된다. 평면상에 주어진  $n$ 개

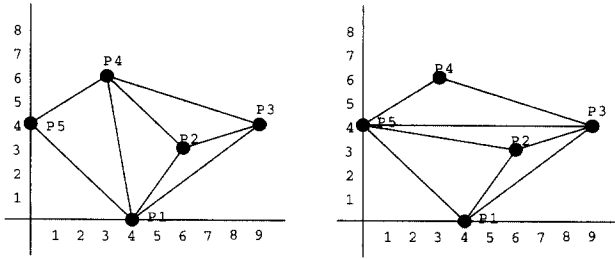


그림 1. 평면상의 점 집합에 대한 삼각화.

의 점 집합을  $C = \{P_1, P_2, P_3, \dots, P_n\}$ , 집합  $C$ 의 모든 점을 연결하는 선분 집합을  $E = \{P_i P_j \mid P_i, P_j \in C, 1 \leq i, j \leq n\}$ ,  $e_i \in E$ ,  $i = 1, \dots, m$ 라 하면, 다음의  $T$ 는 임의의 삼각화를 나타낸다.

$$T = \{e_1, e_2, \dots, e_m\}, T \subseteq E$$

이 때, 삼각화  $T$ 에 대한 가중치  $W(T)$ 를 다음과 같이 정의할 수 있다.  $|e_i|$ 는 선분  $e_i$ 의 가중치를 나타낸다.

$$W(T) = \sum_{i=1}^m |e_i|, e_i \in T$$

앞 절에서 보인 바와 같이 하나의 점 집합에는 다수의 삼각화가 있고, 이러한 삼각화 중에서 최소 가중치를 갖는 삼각화를 결정하는 문제를 MWT 문제라 한다. 평면상에 주어진  $n$ 개의 점 집합  $C$ 가 주어졌을 때 MWT 문제는

$$\begin{aligned} &\text{FIND } T_{\min} \\ &\text{such that } W(T_{\min}) = \text{MIN}\{W(T), \\ &T_i \in \{C \text{에서 가능한 모든 삼각화 집합}\} \end{aligned}$$

기존의 방법으로 그리디 삼각화 방법과 딜러니 삼각화가 널리 알려져 있지만 최적화 방법은 아니라는 것이 증명되어 있다(Kirkpatrick, 1995; Manacher and Zobrist, 1979). 그리디 방법에서는 한 번에 하나의 선분만 삽입하고, 매번 삼각형을 이룰 수 있는지 확인한다. 반복 루프는 삼각화에 포함된 선분의 개수가 삼각화가 가질 수 있는 선분 개수를 모두 채웠을 때 종료한다.  $n$ 개의 점이 있을 경우 삼각화가 갖는 선분의 수는  $n(E) = 3 * n - N_n - 3$ 으로 표시된다.  $N_n$ 은 외곽 볼록 다각형을 이루는 점의 개수를 나타낸다. 딜러니 삼각화는 그 쌍인 보로노이 다이어그램과 함께 계산기하학에서 자주 언급되는 방법이다(Aurenhammer, 1991). <그림 2>는 보로노이 다이어그램을 나타내고 있는데, 다음과 같은 방법으로 보로노이 다이어그램을 생성한다. 점 집합  $C$ 의 임의의 두 점  $P_i, P_j$ 를 연결하는 선분  $P_i P_j$ 을 수직 이등분하는 직선은 영역을 이분하게 된다. 이렇게 나뉘어지는 두 영역 중  $P_i$ 를 포함하는 쪽을  $H(P_i, P_j)$ 라 하고, 한 점  $P_i$ 와 다른 모든 점과의  $H$ 들의 교집합  $V(i)$ 는 볼록 다각형을 이루며 최대  $n-1$ 개의 면을 가진다(Preparata and Shamos, 1985). 즉,  $V(i) = \bigcap_{i \neq j} H(P_i, P_j)$ .

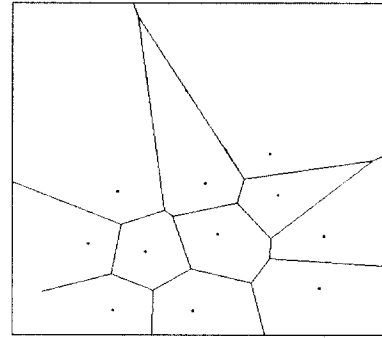


그림 2. 보로노이 다이어그램.

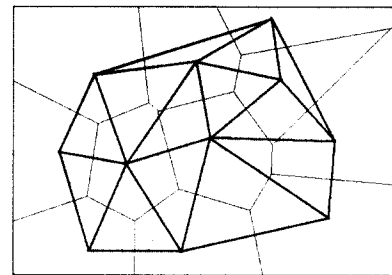


그림 3. 딜러니 삼각화.

딜러니 삼각화는 보로노이 다이어그램에서 서로 인접한 두 점을 연결하는 직선을 모두 연결함으로써 만들어진다. <그림 2>에서 각 점의 보로노이 다각형이 서로 인접한 점들을 연결하는 직선을 그리면 <그림 3>의 굵은 선으로 표현되는데, 이렇게 그려진 도형은 평면을 삼각형 모양으로 분할하며, 서로 교차하지 않는 삼각화가 된다.

### 3. 유전자 알고리즘의 적용

본 장에서는 유전자 알고리즘의 자세한 사항보다는 MWT 문제에 유전자 알고리즘을 적용하는 방안을 염색체구성, 평가연산과 적합함수, 선택연산, 교차연산, 돌연변이연산에 대해 설명하도록 한다. 유전자 알고리즘의 자세한 사항은 Goldberg (1989)에서 발견할 수 있다.

#### 3.1 염색체 구성(Coding)

$n$ 개의 점들로 만들 수 있는 선분의 최대 개수는  $n(n-1)/2$ 이나, 본 연구에서는 무향그래프를 고려하고 있으므로 선분의 연결 상태를 하삼각행렬 또는 상삼각행렬로 표현할 수 있다. 본 연구에서는 하삼각행렬만 저장하여 삼각화의 연결을 나타내며 이 행렬이 염색체를 나타낸다. 이 때, 행렬의 원소는 에지의 연결 여부에 따라 1 혹은 0으로 표현하고 대각원은 저장하지 않는다. 즉, 점  $P_i$ 와 점  $P_j (i > j)$ 가 연결되었다면 행렬의  $i$ 행  $j$ 열의 원소  $m_{ij}$ 는 1이 되고, 연결되지 않았다면  $m_{ij}$ 는 0으로 설

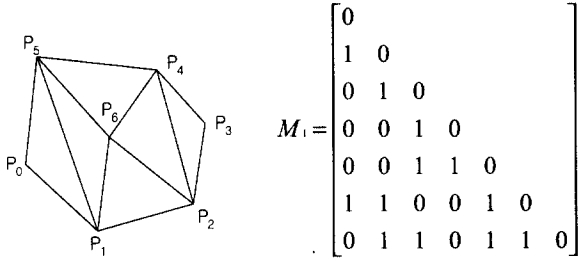


그림 4. 점들의 삼각화.      그림 5. 삼각화의 탐색체 구성.

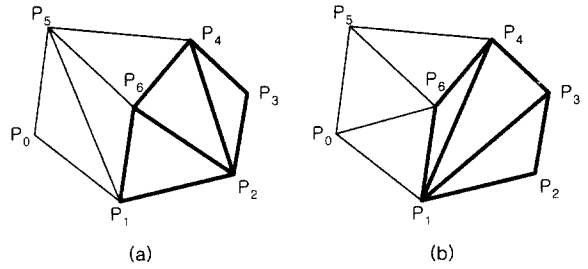


그림 7. 부모 탐색체.

정된다. 하삼각행렬 집합  $M_k$ ,  $k=1, 2, 3, \dots$ 는 한 세대의 탐색체 집합을 나타내며,  $M_k$ 를 한 세대의  $k$ 번째 탐색체라고 정의한다. <그림 4>, <그림 5>는 삼각화의 예와 그 때의 하삼각행렬 구조를 보여 주고 있다(Qin, Wang, and Gong, 1977).

### 3.2 평가연산과 적합함수

한 탐색체의 가중치  $W$ 는  $W = \sum_i \sum_j (l_{ij} \cdot m_{ij})$ 로 정의된다. 공식에서  $l_{ij}$ 는 점  $P_i, P_j$ 를 연결하는 선분  $P_i P_j$ 의 길이이다. 즉, 삼각화의 가중치는 연결되어 있는 선분길이의 총합으로 정의된다. <그림 6>은 간단한 삼각화와 계산된 가중치를 보여 주고 있다.

MWT 문제는 최소 가중치를 갖는 삼각화를 찾는 문제이므로 가중치가 작은 탐색체일수록 적합함수(Fitness Function)값은 커져야 한다. 따라서 탐색체  $k$ 의 적합함수를 다음과 같이 정의한다. 이때  $W_k$ 는 탐색체  $k$ 의 가중치를 나타낸다.

$$f(k) = \frac{1}{W_k}$$

### 3.3 선택연산

유전자 알고리즘을 수행할 때 다음 세대를 구성하기 위하여 부모 세대로부터 탐색체를 선택하는 과정이 필요하다. 이 때 선택연산에 의하여 탐색체를 선택하는데, 본 논문에서는 Roulette Wheel 방법(Goldburg, 1989)을 사용하였다. 한 세대의 모든 탐색체의 적합함수값을 계산하고 백분율로 나타낸 다음,

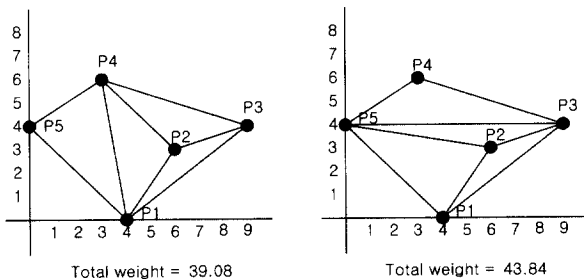


그림 6. 삼각화와 가중치.

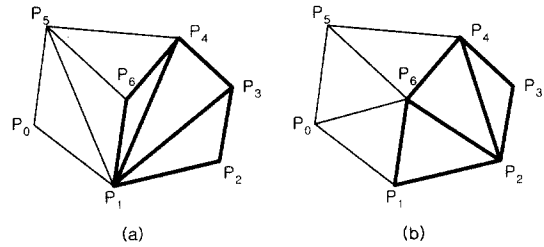


그림 8. 교차연산 수행 후의 자식 탐색체.

난수를 발생시켜 난수가 가리키는 부분의 탐색체를 선택한다. 이 때, 적합함수값이 큰 탐색체가 작은 탐색체보다 선택될 확률이 높아지게 된다. 결국, 적합함수값이 큰 탐색체가 주도적으로 다음 세대를 구성하는 것이다.

### 3.4 교차연산

본 논문에서 연구한 MWT 문제는 문제의 특성상 문제의 정의에 위배되지 않도록 다각형 교차연산을 적용해야 한다. 즉, 선분의 연결모양이 바뀐 후에도 삼각형을 이룰 수 있도록 미리 고려하여야 한다. <그림 7>은 선택연산에 의해 선택된 두 개의 부모 탐색체이고, <그림 8>은 <그림 7>에 대해서 교차연산을 수행한 후의 자식 탐색체이다(Qin, Wang, and Gong, 1977).

이 과정이 실제로 수행되는 과정을 탐색체구성을 이용하여 다음과 같이 나타낼 수 있다.  $m_{3,1}$ ,  $m_{4,1}$ ,  $m_{4,2}$ ,  $m_{6,2}$ 에서 값이 달라지는 것을 관찰할 수 있다.

에지		$m_{1,0}$	$m_{2,0}$	$m_{2,1}$	$m_{3,0}$	$m_{3,1}$	$m_{3,2}$	$m_{4,0}$
교차연산 전 탐색체	(a)	1	0	1	0	0	1	0
	(b)	1	0	1	0	1	1	0
교차연산 후 탐색체	(a)	1	0	1	0	1	1	0
	(b)	1	0	1	0	0	1	0

에지		$m_{4,1}$	$m_{4,2}$	$m_{4,3}$	$m_{5,0}$	$m_{5,1}$	$m_{5,2}$	$m_{5,3}$
교차연산 전 탐색체	(a)	0	1	1	1	1	0	0
	(b)	1	0	1	1	0	0	0
교차연산 후 탐색체	(a)	1	0	1	1	1	0	0
	(b)	0	1	1	1	0	0	0

에지		$m_{5,4}$	$m_{6,0}$	$m_{6,1}$	$m_{6,2}$	$m_{6,3}$	$m_{6,4}$	$m_{6,5}$
교차연산 전	(a)	1	0	1	1	0	1	1
염색체	(b)	1	1	1	0	0	1	1
교차연산 후	(a)	1	0	1	0	0	1	1
염색체	(b)	1	1	1	1	0	1	1

교차연산을 위해 공통 최소다각형을 찾는 알고리즘이 필요 한데, 부모 염색체에 대해 공통 최소다각형 영역을 찾는 알고 리듬을 적용할 수 있다(Qin, Wang, and Gong, 1997). 다음은 공 통 최소다각형 영역을 찾는 알고리즘이다.

초기화: -  $T1, T2$ : 부모 스트링, Stack  $S$ .  
 - 최소다각형 에지 집합  $M = \emptyset$   
 두 부모 중 한쪽 부모에만 있는 선분  $P_iP_j$ 를 무작위로 선택.  
 $P_iP_j$ 를 포함하는 최소 삼각형을 결정.( $P_iP_j$ 를  $T1$ 에 있는 선 분이라고 가정)  
 $S$ 에 최소삼각형의 나머지 두 에지를 push.  
 while  $S$ 가 비지 않는 동안  
    $S$ 에서 에지  $e$ 를 pop.  
   만약,  $e$ 가 양쪽 삼각형에 모두 존재한다면  
      $M \leftarrow M \cup e$ .  
   아니면,  
      $T1$ 에서  $e$ 를 포함하는 최소 삼각형의 나머지 두 선분 을  $S$ 에 push.  
 end while  
 $M$ 은 두 부모 스트링에 공통으로 존재하는 최소다각형.

즉, 교차연산은 두 개의 염색체를 선정하고 공통 다각형을 찾는 알고리즘을 이용하여 서로 교차할 부분을 찾는다. 그 후 서로 교차시킬 부분을 교차시켜서 새로운 자식 염색체를 만드 는 것이다.

3.5 돌연변이 연산

본 논문에서의 돌연변이 연산은 염색체의 임의의 선분을 선택하고 선분이 포함되어 있는 사각형의 다른 쌍의 대각선으로 교체함으로써 수행한다(<그림 9>). 이 때 문제의 정의에 위배 되는 경우는 돌연변이 연산을 수행하지 않는다.

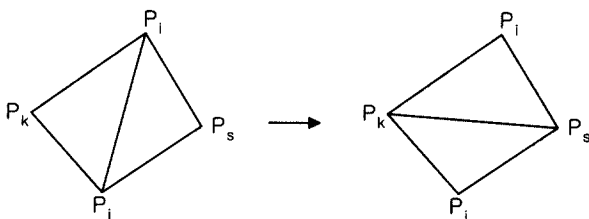


그림 9. 돌연변이 연산.

돌연변이 연산을 염색체구성을 이용하여 다음과 같이 나타 낼 수 있다.  $m_{ij}, m_{ks}$ 에서 값이 달라지는 것을 관찰할 수 있다.

에 지	$m_{ij}$	$m_{ik}$	$m_{is}$	$m_{jk}$	$m_{js}$	$m_{ks}$
돌연변이 연산 전 염색체	1	1	1	1	1	0
돌연변이 연산 후 염색체	0	1	1	1	1	1

하나의 선분을 갖고 그 선분을 대각선으로 하는 사각형을 확인할 수 있는 것은 현 상태에서 삼각화가 이루어져 있으 로 하나의 선분을 끼고 있는 삼각형을 반드시 2개를 찾을 수 있 으므로 가능하다.

4. 문제변형의 정의 및 적용

4.1 문제변형 정의

다단계 탐색공간 스무딩 방법이란 주어진 문제의 해 공간을 원래의 패턴을 지닌 여러 개의 완만한 곡선으로 만들고, 가장 완만한 탐색공간으로부터 시작하여 각 단계의 해 집합을 다음 단계의 초기해로 설정하는 과정을 반복하여 최종적으로 원래 문제의 해 공간에 대한 문제를 해결하는 방법이다. 이러한 다 단계 탐색공간 스무딩 방법의 특징은 처음부터 문제의 해를 직접 해결하지 않는다는 것이다. 즉, 처음에는 전체적인 윤곽 을 잡아 문제의 해를 구하고 점차적으로 원래 문제에 가깝게 변형하면서 해를 구함으로써 최종적으로는 주어진 문제를 풀 게 된다. 이렇게 함으로써 문제해결의 초기단계에서는 굴곡이 완화된 탐색공간을 이용하여 해를 찾으므로 초기에 국부해에 빠져 더 이상 좋은 해를 찾지 못하는 가능성을 줄일 수 있다(강 병천, 한치근, 1998; 이정은, 한치근, 1999).

MWT 문제를 다단계 탐색공간 스무딩 방법을 적용하기 위 해서 스무딩의 대상을 결정하여야 한다. 본 연구에서는 주어 진 문제의 모든 점들의 중심을 구해서 그 중심으로부터 각 점 까지의 거리를 이용하였다. 스무딩의 기본적인 아이디어는 Gu 와 Huang (1994)에서 소개된 것을 사용하였는데, 이들의 연구 에서는 각 에지의 길이를 스무딩하여 문제를 해결하였고, 본 연구에서는 스무딩의 기준이 되는 점을 점 집합의 모든 점들 의 중심점으로 설정하였으며, 중심점으로부터 각 점들의 평균 거리를 반지름으로 하는 가상의 원을 기준으로 스무딩을 수행 하였다.

<그림 10>, <그림 11>은 6개의 점을 이용하여 하나의 동 심원을 그리고, 그것을 변형하는 것을 설명하고 있다. <그림 10>은 문제로 주어진 점들과 중심점으로부터의 평균 거리를 반지름으로 하는 원을 나타낸 것이다. <그림 11>은 다음에 설 명하는 스무딩 함수에 의하여 변형된 점들의 초기 단계이며, 탐색공간은 <그림 11>의 모양으로 시작하여 단계적인 스무

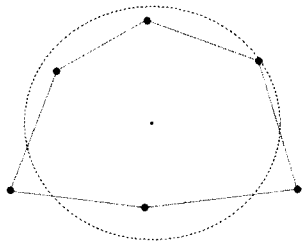


그림 10. 주어진 문제의 점 집합.

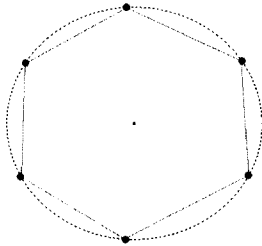


그림 11. 초기 단계의 점들의 위치.

딩에 의하여 <그림 10>의, 즉 원래의 탐색공간으로 변형된다. 이 원래의 탐색공간을 이용하여 주어진 문제의 해를 구할 수 있다.

삼각화에서의 외곽 볼록 다각형은 삼각화를 구성하는 선분의 개수를 결정하는 기준이기 때문에 삼각화의 모양을 결정하는 지표가 된다. 따라서 스무딩할 때 볼록 외곽 다각형의 구성, 즉 삼각화를 구성하는 선분의 개수가 달라지지 않게 하기 위하여 밖으로부터의 외곽 볼록 다각형을 이루는 점들을 차례 차례 대상으로 삼아 각각의 볼록 외곽 다각형에 속한 점들을 스무딩 방법으로 위치 이동을 하게 된다. 그러나 외곽 볼록 다각형으로부터 직접적으로 삼각화를 유도하는 것은 아니고 초기해를 구할 때에 무작위로 삼각화를 시도하게 된다. 무작위로 삼각화를 구하는 방법은 4.2절에서 소개하도록 한다. <그림 12>는 임의의 점 집합에 대한 단계적인 외곽 볼록 다각형을 나타낸다.

점 집합의 모든 점들에 대한  $x$ 좌표와  $y$ 좌표 각각의 평균을 구함으로써 문제 공간의 중심점을 구하고, 각 단계적인 볼록 외곽 다각형에 속한 점들의 중심점으로부터의 거리의 평균을

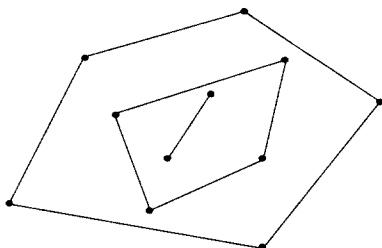


그림 12. 11개의 점 집합과 외곽 볼록 다각형.

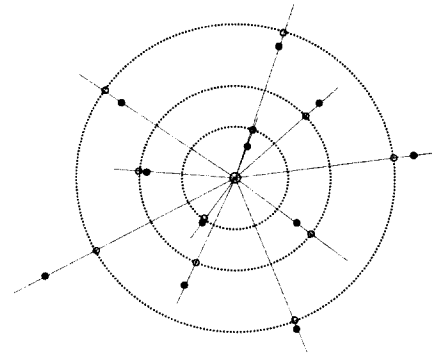


그림 13. 볼록 외곽 다각형의 동심원.

반지름으로 하는 가상의 원을 그리면, 이 때 단계적인 볼록 외곽 다각형의 개수만큼의 동심원을 만들 수 있는데, 모든 점들은 자신이 속한 원주를 중심으로 이동하게 되는 것이다. 결국, 문제 변형의 초기단계에서는 모든 점들은 동심원에 가까운 위치에 있게 되고 알고리즘이 진행됨에 따라 각 점들은 원래의 위치로 이동하게 된다.

<그림 13>에는 3개의 동심원이 있으며, 흰 점은 문제 변형의 초기단계의 점들의 위치를 나타내고 검은 점은 원래 위치를 나타낸다. 외곽 볼록 다각형은 패키지 포장(Package-Wrapping) 알고리즘을 이용하여 찾을 수 있다(Sedgewick, 1992). 패키지 포장 알고리즘에서는 주어진 점들의 좌표를 이용하여 외곽 볼록 다각형을 찾는데, 먼저  $y$ 축으로 가장 작은 값을 갖는 점  $a$ 를 찾는다. 점  $a$ 를 지나는 수평선을 생성하고, 점  $a$ 를 중심으로 시계 반대방향으로 직선을 회전시켜 만나는 점을  $b$ 라 한다. 선분  $ab$ 는 반드시 외곽 볼록 다각형을 구성하는 선분이 된다. 점  $b$ 에서는 계속 직선을 시계 반대방향으로 회전시켜 연속적으로 외곽 볼록 다각형을 구성하는 선분들을 찾는다.

점들의 이동은 다음과 같은 방법으로 계산된다.  $l$ 개의 외곽 볼록 다각형이 있다고 가정할 때, 외곽 볼록 다각형  $k$ 를 이루는 점들이 이들 점들과 중심점과의 평균 거리  $\bar{d}^k$ 에 가까워지도록 하기 위하여 외곽 볼록 다각형  $k$ 상의 점  $i$ 의 스케일링 된 거리를 다음과 같이 정의한다.

$$d_i(a) = \begin{cases} \bar{d}^k + (d_i - \bar{d}^k) \times \alpha & \text{if } d_i \geq \bar{d}^k \\ \bar{d}^k - (\bar{d}^k - d_i) \times \alpha & \text{if } d_i < \bar{d}^k \end{cases}, d_i = |P_c P_i|$$

여기서  $P_c(x_c, y_c)$ 는 문제 공간의 중심점으로서  $x_c, y_c$ 는 주어진 모든 점들의  $x, y$  좌표 각각의 평균이고,  $\bar{d}^k$ 는 문제 공간 중심점으로부터의 외곽 볼록 다각형  $k$ 상의 각 점까지의 거리의 평균을 나타낸다. 또,  $d_i$ 는 공간 중심점으로부터 점  $i$ 까지의 거리이다. 스무딩 인수  $\alpha$ 는 1보다 크지 않는 실수 값을 갖는다.  $\alpha$ 의 값이 1일 때  $d_i(a)$ 는 스무딩되지 않은 거리 값을 갖게 되고, 0에 가까워지면 평균 거리에 가까워진다. 이렇게 계산된 스무딩 거리를 바탕으로 하여 변형된 문제의 점들의 좌표를 계산한다. 점  $P_i(x, y)$ 의 새로운 좌표  $P_i(x(\alpha), y(\alpha))$ 는 다음

석과 같이 주어진다.

$$x(\alpha) = \frac{d_i(\alpha)}{d_i} (x - x_c) + x_c$$

$$y(\alpha) = \frac{d_i(\alpha)}{d_i} (y - y_c) + y_c$$

따라서,  $\alpha$  값을 변화시켜 새로운 탐색공간을 생성할 수 있고, 이전 탐색공간의 해가 새로운 다음 단계의 탐색공간의 초기해로 사용된다.

#### 4.2 문제 변형의 적용

본 논문에서는 기존 유전자 알고리즘의 효율 증대를 위하여 문제 변형을 이용한 다중 유전자 알고리즘의 개념을 적용하였다. 기존의 유전자 알고리즘의 초기해는 일반적으로 무작위 방법으로 생성해 내지만 본 논문의 다중 유전자 알고리즘에서는 각각의 유전자 알고리즘의 초기해를 이전 단계의 유전자 알고리즘을 수행한 결과 세대에서 가져오는데, 다만 첫 번째 알고리즘 수행시에는 무작위 방법으로 초기해를 생성한다. 그 무작위 방법으로 초기해를 구하는 방법은 다음과 같다. 최초의 점집합이 주어지면 두 개의 점을 연결하여 나타낼 수 있는 모든 가능한 선분을 대상으로 무작위로 순서를 정하고, 각 선분을 하나씩 선택하면서 삼각화의 정의에 위배되지 않으면 삼각화의 구성요소로 저장하고 위배되면 버리는 방법으로 삼각화를 하였다.

다음은 본 논문에서 제시한 유전자 알고리즘(EGA: Enhanced GA)의 수행 과정이다.

```

Procedure EGA
  문제변형(  $\alpha = 0$  )
  초기해집합 생성
  While  $\alpha \leq 1$ 
    GA 수행
    문제변형( $\alpha$ )
    해집합 평가
    다음 단계의 초기해 생성
     $\alpha$  증가
  end while
    
```

본 논문의 알고리즘을 수행할 때, 문제의 점들의 좌표가 바뀌므로 삼각화 문제의 특성상, 이전 단계의 유전자 알고리즘을 수행한 결과의 해 집합 중에는 다음 단계의 유전자 알고리즘의 변형된 문제에 적합하지 않은 해가 존재할 수 있다. 적합하지 않은 해란 삼각화의 정의를 위배하는 해로서, 해를 구성하는 선분이 서로 교차하는지를 확인하여 교차하는 선분이 있으면 그 해는 적합하지 않은 해가 된다. 이 때 두 선분이 교차하

는지 확인하는 알고리즘을 사용한다(Sedgewick, 1992). 따라서 각 유전자 알고리즘이 수행된 이후 변형된 문제에 대하여 해 집합을 평가하고, 적합하지 않은 해를 적합한 해로 대체하는 방법이 필요하다. 본 연구에서는 적합하지 않은 해를 삭제하고, 부족한 개체를 보충하기 위해서 무작위 방법으로 적합한 해를 생성하여 보충하는 방법을 사용하였다.

### 5. 실험 및 성능평가

실험에 사용하기 위한 문제는 10, 20, 30, 40개의 점을 각각 5번씩 무작위 방법으로 선정하여 총 20개의 문제를 생성하였다. 이 때, 각 점의 좌표범위는 0~3000 사이의 자연수 값으로 제한하였다. 문제 변형을 위한  $\alpha$  Factor의 값은 초기  $\frac{1}{12}$  부터 1까지  $\frac{1}{12}$  씩 증가시켰다. 즉, 12단계로 문제를 나누어 해결하였다. 각 단계의 유전자 알고리즘의 세대수는 80 세대로 제한하였다. Qin, Wang, Gong(1997)의 기존방법에서는 유전자 알고리즘의 세대수는 1000 세대로 제한하였다. 또한, 한 세대의 개체수는 100개로 설정하여 알고리즘을 수행하였다. 본 논문에서 제시한 방법과 기존 방법의 유전자 알고리즘에서의 교차연산의 확률은 70%, 돌연변이 연산의 확률은 10%로 실험하였다. 이는, 기존 유전자 알고리즘을 사용하였을 때 최상의 결과를 나타내는 확률값으로서, 실험 대상의 문제는 점 10개와 점 20개의 문제를 사용하였다.

<표 1> ~ <표 4>는 실험 결과를 도표로 나타낸 것이다. 각 표에서 가중치는 유전자 알고리즘을 수행한 후 최종 세대의 최소 10% 안에 드는 개체의 평균 가중치 값이고, 시간은 전체 수행 시간을 의미하며 단위는 초이다. 향상비율( (GA-EGA) / GA)은 기존 방법인 GA의 성과와 본 논문에서 제시한 방법인 EGA의 성능을 비율로 나타낸 것이다. 예를 들어 <표 1>의 문제 1을 실험한 결과 본 논문에서 제시한 EGA의 방법이 GA에 비하여 가중치는 0.8%만큼 더 좋은 성능을 보였으며, 시간은 7.4%만큼 더 단축되었다는 뜻이다.

<표 1> ~ <표 4>의 결과를 보면 제안한 방법이 기존 방법에 비하여 전체 문제 20개 중 17개에서 가중치의 향상을 보였고, 8개의 시간 향상을 보였다. 음영을 준 부분이 EGA가 더 좋은 결과를 준 경우를 표시하고 있다. 시간이 더 소요되고 있

표 1. 10개의 점으로 구성된 문제의 실험 결과

문제 번호	EGA		GA		EGA의 향상비율(%)	
	가중치	시간	가중치	시간	가중치	시간
1	22036	125	22202	135	0.8	7.4
2	25109	135	25045	139	-0.3	2.9
3	22809	166	22923	182	0.5	8.8
4	21408	195	21930	200	2.4	2.5
5	16860	146	16979	144	0.7	-1.4

표 2. 20개의 점으로 구성된 문제의 실험 결과

문제 번호	EGA		GA		EGA의 향상비율(%)	
	가중치	시간	가중치	시간	가중치	시간
1	34024	413	34151	417	0.4	1.0
2	38503	402	39670	416	2.9	3.4
3	39927	460	39699	423	-0.6	-8.7
4	39164	434	39675	430	1.3	-0.9
5	42473	409	39257	415	-8.2	1.4

표 3. 30개의 점으로 구성된 문제의 실험 결과

문제 번호	EGA		GA		EGA의 향상비율(%)	
	가중치	시간	가중치	시간	가중치	시간
1	45944	954	48448	906	5.2	-5.3
2	47551	917	47850	825	0.6	-11.2
3	53490	895	61202	902	12.6	0.8
4	53616	1005	55693	999	3.7	-0.6
5	55031	1067	57272	883	3.9	-20.8

표 4. 40개의 점으로 구성된 문제의 실험 결과

문제 번호	EGA		GA		EGA의 향상비율(%)	
	가중치	시간	가중치	시간	가중치	시간
1	69304	2016	75454	1508	8.2	-33.7
2	67469	1902	69002	1575	2.2	-20.8
3	74015	2011	74896	1560	1.2	-28.9
4	72213	1933	73055	1601	1.2	-20.7
5	74789	2007	76499	1522	2.2	-31.9

나 가중치에서는 평균적으로 2%의 향상을 가져왔다.

문제의 크기가 증가할수록 EGA의 수행시간이 GA의 수행시간에 비해 많이 소요되는 것을 관찰할 수 있었다. 이것은 EGA인 경우 외곽 다각형을 구하는 시간이 추가되는데,  $n$ 개의 점이 있을 경우 패키지 포장 알고리즘은  $O(n^2)$ 의 시간이 소요된다. 따라서  $n$ 이 증가할수록 시간의 차이가 늘어나게 된다.

## 6. 결론

본 논문에서는 MWT 문제에 유전자 알고리즘을 적용시킬 때 점의 위치를 변경시키는 방법을 이용하여 보다 좋은 해로의 수렴을 유도하였다. 스케일링 값을 조절하여 주어진 문제의 점의 위치를 변경시키고, 짧은 세대의 유전자 알고리즘을 중복하여 수행함으로써 무작위로 흩어져 있는 점들의 위치 관계를 개략적으로 분석하며 점차 세부적으로 연결 관계를 유도해 내는 효과를 보였다.

기존의 유전자 알고리즘에 비해서 제안된 방법은 해에서는 더 좋은 결과를 보였지만, 시간에서는 중간 단계의 계산량이 많아 더 많은 시간이 필요한 것을 관찰할 수 있었다. 본 연구의 결과를 3차원 공간의 점들에 대해서도 적용할 수 있다. 결과를 3차원 공간으로 확장하기 위해서는 3차원 공간상의 점 좌표들을 변형하는 방법이 필요한데, 본 논문에서 사용한 단계적인 외곽 다각형을 외곽 다면체로 확장하고, 중심원의 개념을 동심구로 변형하여 문제를 풀어 나갈 수 있을 것이다.

## 참고문헌

강병천, 한치근 (1998), 확장된 집합커버문제 해결을 위한 유전자 알고리즘, *한국정보과학회논문지(A)*, 25(2), 198-204.

이정은, 한치근 (1999), T-쿼라링 문제를 위한 탐색공간 스무딩 Simulated Annealing 방법, *대한산업공학회지*, 25(2), 226-232.

Aurenhammer, F. (1991), Voronoi Diagrams - A Survey of Fundamental Geometric Data structure, *ACM Computing Surveys*, 23(3), 345-405.

Garey, M. R., and Johnson, D. S. (1979), *Computers and Intractability - A Guide to the Theory of NP-Completeness*, Freeman, New York.

Goldburg, D. E. (1989), *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Publishing Company, Inc.

Gu, J., and Huang, X. (1994), Efficient Local Search with Search Space Smoothing: A Case Study of the Traveling Salesman Problem(TSP), *IEEE Transactions on Systems, Man, and Cybernetics*, 24(8), 728-735.

Kirkpatrick, D. G. (1980), A Note on Delaunay and Optimal Triangulations, *Inform. Process. Lett.*, 10(3), 127-128.

Manacher, G. K., and Zobrist, A. L. (1979), Neither the Greedy Nor the Delaunay Triangulation of a Planar Point Set Approximates the Optimal Triangulation, *Inform. Process. Lett.*, 9(1), 31-34.

Preparata, F. P., and Shamos M. L. (1985), *Computational Geometry*, Springer-Verlag, New York Inc.

Qin, K., Wang, W., and Gong, M. (1997), A Genetic Algorithm for the Minimum Weight Triangulation, *Proceedings of the 1997 IEEE Conference*, 541-546.

Sedgewick, R. (1992), *Algorithms in C++*, Addison-Wesley Publishing Company, Inc.