韓國經營科學會誌
第23卷 第1號
1998年 3月

유전자 알고리듬과 K-평균법을 이용한 지역 분할

임동순* · 오현승**

Zone Clustering Using a Genetic Algorithm and K-Means*

Dong-soon Yim* · Hyun-seung Oh**

Abstract

The zone clustering problem arising from several area such as deciding the optimal location of ambient measuring stations is to divide the 2-dimensional area into several sub areas in which included individual zone shows similar properties. In general, the optimal solution of this problem is very hard to obtain. Therefore, instead of finding an optimal solution, the generation of near optimal solution within the limited time is more meaningful. In this study, the combination of a genetic algorithm and the modified k-means method is used to obtain the near optimal solution. To exploit the genetic algorithm effectively, a representation of chromosomes and appropriate genetic operators are proposed. The k-means method which is originally devised to solve the object clustering problem is modified to improve the solutions obtained from the genetic algorithm. The experiment shows that the proposed method generates the near optimal solution efficiently.

1. 서 론

지역 분할 문제는 2차원 공간상의 지역을 특성치가 서로 상이한 부분지역으로 분할하는 문제이다. 이 문제는 널리 알려진 개체분할 문제[1,4,7,12]와 비슷하나, 2차원 공간상에서 같은 그룹에 속한 개체들은 서로 인접, 연결되어 있어야 하는 조건을 갖고 있다. 예를 들면, 대기오염 측정소의 최적 위치를 선정하기 위하여 고

* 한남대학교 산업공학과
** 한남대학교 산업공학과

려되는 지역을 서로 이질적인 오염도 특성을 나타내는 부분 지역으로 분할한 후 각 부분 지역에 대하여 오염도를 대표할 수 있는 지점에 측정소를 위치하는 방법을 사용할 수 있다[8,9]. 즉, 전체 지역을 격자 모양으로 구분하여 각 격자들 간의 오염도 분포에 대한 유사성을 표현하는 계수를 구한다. 이러한 유사성계수를 이용하여 전체지역을 서로 다른 오염도 특성을 갖는 부분지역으로 분할한다.

이질적인 특성을 나타내는 부분 지역으로의 분할을 위해서는 이에 알맞은 목적함수를 결정하여야 한다. 결정된 목적함수에서의 지역분할은 제한된 시간 내에 최적해를 쉽게 구할 수 없게 된다. 이러한 문제의 해결을 위해서는 최적해는 아니지만 제한된 시간 내에 근사해를 구할 수 있도록 하는 것이 중요하다. 근사해를 구할 수 있는 한 방법으로서 유전자 알고리듬을 고려할 수 있다. 그러나, 지역 분할 문제를 해결하기 위하여는 이에 적합한 유전자 암호화 방법과 유전자 연산자의 정의가 필요하다. 유전자 알고리듬에 의한 해는 대략적인 근사해로서 보다 좋은 부분 최적해(local optimum)로 변화 시키기 위한 부가적인 절차를 필요로 한다. 부분 최적화 방법으로는 개체 분할에 적용되는 K-평균 방법을 고려할 수 있으나, 지역 분할 문제에 적용될 수 있도록 수정되어야 한다. 본 연구에서는 수정된 K-평균 방법을 유전자 알고리듬의 해에 적용하여 해의 향상을 갖고 오도록 하였다.

2. 지역 분할 문제

지역 분할 문제를 해결하기 위하여, 고려되는 지역을 격자모양의 그래프 (graph)로 표현한 후 특성치가 유사한 인접 노드(node)들을 같은 그룹으로 묶어 전 지역을 주어진 수 만큼의 그룹으로 분할하는 방법론을 이용한다. 즉, 그래프를 주어진 수만큼의 부분 그래프 (sub graphs)로 분할한다. 두 노드간의 특성치에 대한 유사성을 결정하는 척도로서는 여러 가지가 있을 수 있으나, 본 연구에서는 문제를 간단히 하기 위하여 인접한 두 노드에서 특성치의 차이를 유사성의 척도로 간주한다. 즉, 특성치의 차이가 작을수록 유사성이 크다. 또한, 특성치는 하나의 값으로 표현된다고 가정한다. 특성치가 유사한 노드들을 같은 그룹으로 묶는 문제의 정의는 여러 경우를 고려할 수 있다. 그 중 하나는 다음과 같다.

문제 1

$$\text{Minimize} \sum_{k=1}^{M}\left(\sum_{j\in G_k}(x_j-\bar{x}_k)^2\right)$$

subject to) Adjacency constraint

x_j는 j 번째 노드에서의 특성치를 의미한다. G_k를 k 번째 그룹에 속한 노드들의 집합이라고 할 때 $\overline{x_k}$는 G_k에 속한 노드들의 특성치 평균값을 의미한다. 즉,

$\overline{x_k} = \sum_{j\in G_k} x_j / N(G_k)$, $N(G_k)$는 그룹 G_k 에 속한 노드의수.

위의 목적함수는 주어진 M개 그룹에 대해 각 노드의 특성치와 노드가 속한 그룹평균의 오차제곱 합을 최소화 한다. 즉, 각 그룹 내 노드들의 특성치에 대한 흩어짐 정도를 최소화 한다. N개의 노드들을 M개의 그룹으로 분할 하는데 있어 각 그룹에 속한 노드들은 인접조건

(adjacency constraint)을 만족하여야 한다. 인접조건은 한 그룹내의 모든 노드는 서로 인접해야 하는 제약조건을 나타내며 다음과 같이 정의된다.

인접조건

노드 ($n_1, n_2, \cdots n_N$)와 edge들로 구성된 그래프가 있을 때 그룹, $G_k(k=1,2,\cdots,M)$에 속하는 어느 두 노드, n_s, n_t, 사이에 다음과 같은 path가 존재하여야 한다.

($n_s, n_1, n_2 \cdots, n_p, n_t$),

단, $n_1, n_2, \cdots, n_p \in G_k$

특성치가 유사한 노드들의 그룹핑을 위한 또 다른 문제의 정의는 다음과 같다.

문제 2

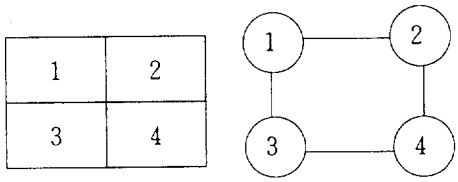
$$\text{Maximize} \sum_{k=1}^{M} N(G_k)(\bar{x}_k - \bar{x})^2$$

subject to) Adjacency constraint

이 문제는 각 그룹의 평균 ($\overline{x_k}$)과 전체평균 ($\bar{x}$)의 차이에 대한 제곱합을 그룹에 속한 노드의 수 만큼 가중치를 두어 흩어짐을 구한 것이다. 이 흩어짐을 최대화하여 각 그룹에 대한 이질성을 최대화한 것이다. 인접조건을 갖는 지역분할에서 이 목적함수는 나쁜 결과를 갖고 올 수 있다. 서로 지역적으로 멀리 떨어진 두 그룹의 평균이 같은 경우에 위 문제 하에서는 나쁜 해 이지만, 두 그룹 중간 지역의 특성치 상이로 인하여 두 그룹이 분리 되는 것이 바람직할 수 있기 때문이다. 이런 이유로 인하여 본 연구에서는 문제 2 보다는 문제 1하에서의 최적해를 구하는 것을 대상으로 한다.

예제

위 문제를 설명하기 위하여 다음의 간단한 예제를 소개한다. 어느 지역을 [그림 1]과 같이 격자로 구분 했을 때 격자에 의한 각 분할 지역을 노드로 표현하고 인접한 격자(노드)들을 edge로 잇는 그래프로 표현할 수 있다. 노드들을 2개 그룹으로 묶는다고 가정하자. 노드에서의 특성치는 각각 4, 1, 10, 8 이다. 인접조건을 만족하는 해의 수는 6가지가 있으며, 각 해와 목적함수 값은 〈표 1〉과 같다. 〈표 1〉은 전체 노드를 노드 (1,2) 와 (3,4)로 구성되는 두개 그룹으로 구분할 때 문제1의 경우 목적함수 값은 6.5로 6개의 대안중 최적해임을 보여준다. 문제 2의 경우에도 문제1에서의 최적해와 일치한다.



(a)격자표현 (b) 그래프 표현

[그림 1] 격자와 그래프 표현

〈표 1〉 지역분할 대안 비교

대안 번호	그룹 1	그룹 2	목적함수값 (문제1)	목적함수 값 (문제2)
1	1	2,3,4	22.59	4.08
2	2	1,3,4	18.67	30.08
3	3	1,2,4	24.67	24.08
4	4	1,2,3	42	6.75
5	1,2	3,4	6.5	42.25
6	1,3	2,4	42.5	6.25

위에 정의된 최적화 문제는 인접조건을 고려하지 않을 경우, 다음 장에서 설명될 개체 분할 문제와 동일하다. n개의 개체를 m개의 그룹으

로 분할하는 개체 분할에서,

$\frac{1}{m!}\sum_{j=1}^{m}(-1)^{m-j}\ {}_mC_j j^n$개의 대안 수가 존재한다. 만약, 100개 개체를 5개 그룹으로 분할한다면 10^{68}개 정도의 대안이 있다[1]. 인접조건을 고려한 문제인 경우에는 이러한 대안 중에서 인접조건을 만족하는 대안만을 선별, 비교함으로써 최적해를 구할 수 있다. 그러나, 노드 수가 증가함에 따라 대안의 수가 기하적으로 증가함으로 문제의 정확한 해 보다는 제한된 시간내에 근사해를 제공하는 것이 더욱 의미가 있다.

3. 개체 분할 문제

전 장에서 정의된 문제는 그룹에 속한 노드들의 인접조건을 제외하면 개체 분할 문제와 같이 단순히 데이터를 군집화(clustering)하는 문제와 일치한다. 일반적으로 데이터를 군집화 하는 방법은 크게 계층적 접근방법(hierarchical approach)과 최적화 접근방법(optimization approach)으로 나뉠 수 있다. 계층적 접근 방법은 각 개체 또는 부분 그룹들을 단계적으로 합하여 최종분류를 얻는 방법과 전체개체를 한 그룹으로 묶은 후 단계적으로 분리하여 최종분류를 구하는 방법으로 나뉠 수 있다. 최적화 접근방법은 초기에 k개의 그룹으로 각 개체를 그룹화 한 후 목적함수의 최적화를 위하여 각 개체를 이동시킨다. 두 접근방법에서 가장 빈번히 이용되는 방법들과 개체 분할 문제에 적용되는 유전자 알고리듬을 설명한다.

3.1 Ward 의 방법

계층적 접근방법에 속하는 이 방법은 그룹내의 분산(ESS: Error Sum of Squares)을 최소화하기 위하여 고안되었다[12]. 즉, 전 장에서의 인접 제한조건을 제외한 문제1을 해결한다. 알고리듬의 첫번째 단계에서는 각 노드가 각기 고유의 그룹에 속해 있다. 즉, 노드 수 만큼의 그룹 수가 존재하여 ESS는 0이다. 이 ESS가 최소로 증가하도록 두개의 노드 또는 그룹들을 합치는 과정을 반복한다. 이 방법은 상대적으로 같은 수의 노드를 포함하는 그룹의 분할을 초래한다.

3.2 K-평균 방법

최적화 접근방법에 속하는 이 방법은 알고리듬의 초기에 전체 노드를 주어진 그룹 수 k개로 분할한 후 각 그룹의 평균을 계산한다. k개 그룹으로의 초기 분할은 임의로 하거나, 계층적 접근방법에 의하여 실행할 수 있다. 다음에는 각 노드들을 조사하여 다른 어떤 그룹평균과의 차이가 그 자신이 속한 그룹의 평균보다 작으면 그 다른 그룹으로 이동시킨다. 다시 그룹평균을 구하고 이 과정을 안정상태에 도달할 때 까지 반복한다. 각 노드들을 조사할 때 한 단계에서 새로이 생성된 그룹평균의 재계산 없이 모든 노드들에 대한 이동여부를 결정하는 방법과 하나의 노드가 이동 될 때마다 각 그룹의 평균을 재계산하는 방법이 있다.

3.3 유전자 알고리듬

유전자 알고리듬은 여러 개의 개체가 동시에

병렬적으로 주어진 환경에 따라 적자생존의 방법으로 진화하여, 궁극적으로 최적의 상태에 도달하는 생태계의 진화이론에서 도입되었다[6]. 이 알고리듬은 여러 개의 개체로 구성된 군집이 진화할 때 구 세대가 얻은 환경에 대한 정보는 염색체에 저장되어 다음세대로 전달된다. 이 때 조상의 염색체가 그대로 복제되어 자손에게 전달되는 것이 아니라 조상의 염색체에 교차(crossover), 돌연변이(mutation), 전위(inversion) 등의 연산을 가하여 얻은 염색체로 전달된다. 구 세대 중에서 한 개체가 선택되어 자손에게 유전정보를 남길 확률은 일반적으로 그 개체가 주위환경, 그리고 나머지 개체와 어떻게 상호 작용하는 가에 의존하는 개체 값(fitness)에 따라 변한다. 일반적으로 개체 값이 좋을수록 자손을 남길 확률이 높아지는 적자생존의 법칙이 적용된다. 이 알고리듬은 여러 가지 종류의 최적화 문제에 응용되어 좋은 결과를 낳고 있다.

유전자 알고리듬을 개체 분할 문제에 적용하기 위하여는 이 문제의 해결에 적합한 염색체의 표현방법과 이 표현에 알맞은 교차, 돌연변이, 전위연산자 등을 정의하여야 한다. 일반적으로 다음에 설명될 개체 분할 문제에 적용되는 유전자 알고리듬 방법들이 있다.

3.3.1 그룹 번호 표현 방법

노드들이 속한 그룹번호를 문자열로 표현하는 매우 단순한 방법이다. 즉, n개의 개체가 있을 경우 이를 k개의 그룹으로 분할하는 표현방법은 $(i_1, i_2, \cdots i_n)$이다. 여기서 j번째 정수 i_j는 j번째 개체가 속한 그룹번호를 의미한다. 〈표 1〉의 5번째 대안은 (1, 1, 2, 2)로 표현된 수 있다. 그룹번호 표현 방법은 표준적인 돌연변이, 교차 등의 연산자 사용을 가능케 한다. 즉, 돌연변이는 랜덤으로 선택된 i_j를 1부터 k 사이의 임의의 수로 바꿔 줄 수 있다. 표준적인 교차 연산자는 언제나 그룹으로 분할 하는 표현방법에 맞는 자손을 생성한다. 그러나, [7]에서도 언급했듯이 이들의 연산자에 의한 자손은 k개 이하의 그룹을 가져올 수 있다. 또한, 같은 분할 결과를 가져오는 두 부모로 부터의 자손은 두 부모의 다른 그룹번호 체계로 인하여 완전히 다른 결과가 될 수 있어 이를 수정할 수 있는 복구 알고리듬이 요구된다.

3.3.2 Bit 표현방법

개체 분할은 노드와 edge로 구성된 그래프로 표현될 수 있고, 이 그래프에서 edge들을 bit로 표현할 수 있다. 두 인접한 노드들이 같은 그룹에 속한다면 노드들을 잇는 edge의 값은 1 이고, 그렇지 않으면 0 이다. 전 절의 예제에서 5번째 대안의 연결된 edge는 (1, 2), (3, 4) 이다. 만약 edge 리스트를 {(1, 2), (1, 3), (2, 4), (3, 4)} 의 순서로 한다면 염색체는 (1001)로 표현된다. 이 표현방법에 대하여는 전형적인 연산자를 사용할 수 없다. 교차 나 돌연변이 등의 연산자에 의한 자손은 유효한 분할이 될 수 없는 경우가 발생할 수 있기 때문이다.

3.3.3 서수화(Ordered) 표현방법

이 표현방법에서 각 분할은 개체들의 순열로 표현되며, 인접한 개체들은 그렇지 않은 개체보다 더 유사하다고 간주된다. 때문에, 해를 직접적으로 제공하기 보다는 개체간 유사성의 정보만을 갖고 있다. 예를 들어, 1부터 6까지의 정수로 구성된 개체들에 대한 한 염색체가 (2, 3, 5,

1, 6, 4)의 순열로 표현됐다고 하자. 개체 2를 포함하는 두개의 개체를 한 그룹으로 묶는다면, 개체 3이 선택된다. 개체 2에 가장 유사한 개체는 인접한 개체 3이기 때문이다. 만약, 이 6개의 개체를 3개의 그룹으로 나눈다면 여러 가지 가능성이 있다. 그 중의 몇 가지는 {2, 3, 5,} {1} {6, 4}, {2} {3, 5, 1}{6, 4} 등이다. 이러한 모든 가능한 해 중에서 가장 좋은 해를 결정하여야 하는 부가적인 계산 절차를 필요로 한다. Fisher[4]등은 가능한 해 중에서 가장 좋은 해를 구하기 위하여 동적 프로그래밍을 이용하였다.

4. 제안된 유전자 표현 방법 및 연산자

4.1 Edge 우선순위 표현방법

전 장에서 설명된 개체 분할문제에 적용되는 표현 방법들은 지역분할 문제에 그대로 적용되기 어렵다. 우선적으로, 개체 분할 문제에 비해 지역분할은 그룹 내에 속한 노드들이 서로 인접해야 한다는 인접조건을 만족하여야 하기 때문이다. 인접조건으로 인하여 표준적인 유전자 연산자의 사용이 불가능해 진다. 가능한, 표준적인 연산자의 사용이 허락되고, 연산자에 의한 자손들은 항상 유효한 해를 생성할 수 있는 방법이 필요케 된다. 본 연구에서 제안되는 표현방법을 위해서는 bit 표현방법에서 이용된 것과 같은 edge 리스트를 구성한다. 그러나, 각 edge에 대해 연결여부를 bit로 표현하는 방법 대신에, 각 edge의 연결 가능성에 대한 우선순위를 부여하는 표현방법을 사용한다(본 연구에서 고안된 이 표현 방법을 edge 우선순위 표현방법으로 부르기로 한다) . 즉, 하나의 edge 우선순위 표현에서 우선순위가 높은 edge로 연결된 두 노드는 우선순위가 낮은 edge로 연결된 두 노드 보다 같은 그룹으로 묶어질 가능성이 크다. 이 표현 방법은 단지 edge로 연결된 두 노드들의 그룹화에 대한 우선 순위 정보만을 갖고 있는 이유로, 개체 분할 문제에서의 서수화 표현 방법에서와 같이 해를 직접적으로 생성하지 못한다. 때문에, 한 염색체가 항상 유효한 하나의 해를 생성할 수 있도록 특별한 복호화(decoding) 절차가 요구된다. [그림 1]의 예에서 모든 edge에 대한 리스트를 {(1, 2), (1, 3), (2, 4), (3, 4)}의 순으로 하고, 각 edge에 대한 우선순위를 1부터 edge의 개수인 4까지의 정수를 할당하여 (1, 3, 4, 2) 로 표현하였다고 하자. 그룹의 수가 2가 될 때까지 우선순위에 의해 edge를 선택하여 그 edge에 연결된 두 노드를 같은 그룹으로 묶는다면, 우선순위가 1, 2인 edge들 (1, 2)와 (3, 4)가 선택되어 〈표 1〉의 최적해 대안 5의 결과를 낳게 된다. 그러나, 우선순위에만 의해 edge를 선택한다면, 하나의 노드만으로 그룹을 형성하는 경우가 빈번히 발생할 수 있다. 만약, 유전자 연산에 의한 자손들에 대한 해 역시, 하나의 노드만으로 구성된 그룹만을 다수 생성한다면, 부분적인 최적해에 빠져 전체 최적해에 근사한 해를 구할 수 없게 된다. 이러한 단점을 보완하기 위하여 edge 우선순위 표현방법을 해로 복호화할 때 다음의 알고리듬에 의한다.

복호화 알고리듬은 edge의 우선순위 합을 최소화하는 최소 걸침 나무(minimal spanning tree)를 구한다. 이 트리에서 M-1개의 edge를 제거하여, 그룹의 수 M 만큼의 부분 트리를 구하면, 각 부분 트리는 유효한 해가 된다. N개의 노드를 갖는 트리에서 M-1개의 edge를 제거하

는 경우의 수는 $_{N-1}C_{M-1}$이다. 본 연구에서는 우선순위가 작은 edge들을 제거 하였다. 이 복호화 알고리듬의 복잡성은 최소 걸침 나무를 구하는 알고리듬의 복잡성에 의존하며, 노드의 수가 n일 때 $O(n^2)$이다.

알고리듬 : Edge 우선순위 표현방법의 복호화

입력 : Edge 우선순위 표현, 노드 수 N,
그룹 수 M

출력 : M 개의 노드 집합

처리 :

단계 0 : Edge의 우선순위 합을 최소화하는 최소 걸침 나무를 구한다.

Step 1 : 최소걸침나무에 속하는 edge들을 우선순위의 증가 순으로 정렬한다.

Step 2 : (M-1개의 제거될 edge들을 구한다.) 정렬된 edge 중 첫번째부터 M-1개의 edge를 제거한다.

Step 3 : 정렬된 edge 리스트의 edge들을 이어 M개의 부분 트리 (노드집합)를 구한다.

4.2 유전자 연산자

Edge 우선순위 표현방법에 적용될 수 있는 교차 연산자는 외판원 문제에 적용되는 path 표현방법에서의 교차 연산자인 PMX (Partially-mapped), OX (Order), CX (Cycle)등을 사용할 수 있다. PMX는 Goldberg 와 Lingle [5]에 의해 제안된 연산자로 한 부모로부터 연속된 부분 유전자를 유전 받고, 다른 부모로부터 가능한 많은 유전자를 상속 받도록 한다. OX[3]는 한 부모로부터 연속된 부분 유전자를 상속 받고, 다른 부모로 부터는 상대적인 순서를 유지하도록 한다. CX[10]는 부모의 절대적인 순서를 유지하도록 자손을 생성케 한다.

이 중 상대적인 순서에 중요성을 둔 OX는 지역 분할 문제에 있어서는 좋은 결과를 가져올 수 없다. 지역 분할에서 우선순위의 상대적인 순서에 대한 유지는 중요성이 없기 때문이다. CX와 PMX는 지역분할 문제의 edge 우선순위 표현 방법에 사용할 수 있으나, CX는 외판원 문제와 같은 순서화 문제에서 일반적으로 PMX에 비해 성능이 떨어진다[11]. 때문에, 본 연구에서는 교차 연산자로서 기존의 PMX방법을 고려하고, 부가적으로 RRX (Relative Rank-based Crossover) 방법을 제안한다. 또한, edge 우선순위 표현방법에 이용될 수 있는 전위 연산자를 설명한다.

PMX

예를 들어 두 부모의 edge 우선순위 표현이 다음과 같다고 하자 (두 개의 임의의 컷 포인트를 | 로 표현하였다).

$p_1 = (1\ 2\ 3\ |\ 4\ 5\ 6\ 7\ |\ 8\ 9)$
$p_2 = (4\ 5\ 2\ |\ 1\ 8\ 7\ 6\ |\ 9\ 3)$

우선 두부모의 컷 포인트사이의 수들은 서로 교환되어 자손에 상속된다.

$o_1 = (x\ x\ x\ |\ 1\ 8\ 7\ 6\ |\ x\ x)$
$o_2 = (x\ x\ x\ |\ 4\ 5\ 6\ 7\ |\ x\ x)$

이 교환으로부터 1-4, 8-5, 7-6, 6-7 의 mapping이 정의된다. 두 자손에서 미결정된 위치에서의 값은 부모의 같은 위치의 값을 상속받는다. 만약, 같은 위치에 있는 부모의 값이 이미 할당되었다면 그 수에 mapping된 수를 대

신 갖도록 한다. 즉, 완성된 두 자손은 다음과 같다.

o_1=(4 2 3 | 1 8 7 6 | 5 9)
o_2=(1 8 2 | 4 5 6 7 | 9 3)

RRX (Relative Rank-based)

RRX 방법은 PMX와 같이 임의의 두 컷 포인트 사이의 연속된 부분 유전자를 자손에 상속한다. 그러나, 나머지 미 결정된 위치의 값은 각 위치의 상대적인 우선순위를 만족하도록 할당한다. PMX에서 언급된 예에서와 같이 교환에 의한 두 자손이 있을 때 미 결정된 위치의 상대적인 순위는 다음과 같다.

o_1=(1 2 3 | a a a a | 4 5)
o_2=(3 4 1 | a a a a | 5 2)

미 할당된 수들을 이 상대적인 우선순위에 의해 할당하면 다음과 같이 완성된 자손을 생성한다.

o_1=(2 3 4 | 1 8 7 6 | 5 9)
o_2=(3 8 1 | 4 5 6 7 | 9 2)

전위 연산자

Edge 우선순위 표현방법에서는 전형적인 돌연변이 연산자를 적용하기가 불가능 하다. 대신에 전형적인 전위 연산자의 사용은 매우 간단히 적용될 수 있다. RRX연산자의 예에서 생성된 첫번째 자식에 대하여 전위연산자를 적용하면, 임의의 두 컷 포인트가 2 와 5 일 때 다음과 같이 2번째 와 5번째 사이의 수들이 전위된다.

o_1=(2 | 8 1 4 3 | 7 5 9)

4.3 부분 최적화 방법에 의한 유전자 알고리듬 해의 향상

유전자 알고리듬에 의한 해를 보다 좋은 해로 변환시키기 위하여 각 개체에 적용하거나, 또는 각 세대의 가장 좋은 해에 부분 최적화 알고리듬(local optimization algorithm)을 적용할 수 있다. 부분 최적화 알고리듬으로는 개체 분할 문제에 적용되는 최적화 알고리듬을 고려 할 수 있으나, 지역 분할 문제에 적용하기 위하여는 노드들의 인접조건을 고려한 수정이 필요하다. 본 연구에서는 유전자 알고리듬에 의한 해를 수정된 K-평균법 방법에 의해 보다 향상된 결과를 가져오도록 하였다. 수정된 K-평균법 알고리듬은 다음과 같다.

인접조건을 고려한 K-평균법 알고리듬

```
Input : (초기해)
        CLUS(I), I=1, … ,N /* 노드 i가 속
        해 있는 그룹번호 */
        X(I), I=1,…,N /* 노드 i의 특성치 */
        그래프 G(N, A) /* N: 노드 집합,
        A: Edge 집합 */
Output : (새로운 해)
         CLUS(I), I=1,… ,N /* 노드 i의 새
         로운 그룹번호 */
Process :
SUCCESS = NO
WHILE (SUCCESS == NO)
{
 SUCCESS = YES
 각 노드(I)에 대하여
 {
  I의 특성치와 I가 속한 그룹평균의 절대차
  이(D0)계산
  WORK = STAY
  노드의 각 인접한 노드(J)에 대하여
   {
```

```
만약 CLUS(I) != CLUS(J) 이면
 {
 I의 특성치와 J가 속한 그룹평균의 절대
 차이(D1) 계산
   만약 D1 < D0 이면
   {
    D0 = D1
    WORK = MOVE
   }
 }
}
만약 Work == MOVE 이면
{
  OLD = CLUS(I)
  CLUS(I) = CLUS(J)
  만약 새로운 해에서 각 그룹의 노드가
  인접했다면
  {
     SUCCESS = NO
     Break
   }
   아니면
     CLUS(I) = OLD
   }
 }
}
```

이 알고리듬의 복잡성은 새로운 해에서 각 그룹의 노드가 인접했는지를 조사하는 횟수에 의존한다. 인접조사는 그래프에서 그룹번호 i 에 속한 노드들을 선택하고, 선택된 노드들간의 edge만을 고려한 그래프에서 깊이 우선 탐색(Depth-First-Search) 방법에 의해 그래프의 연결상태를 조사 할 수 있다. 모든 그룹번호에 대하여 그래프가 연결됐다면 인접조건을 만족한 해라고 할 수 있다. 이 깊이 우선 탐색에 의한 인접조사의 복잡성은 그래프의 전체 edge 수에 비례하여 r 행과 c열을 갖는 지역의 그래프(노드 수 n은 r×c) 에서 O(n) 이다. 인접조사를 수행하는 횟수는 해의 구조에 의존하여 정확한 수는 계산할 수 없다. 주어진 해가 완벽하여 더 이상의 부분 최적화가 필요 없는 경우는 0 이지만 최악의 경우에는 n^2 번 정도의 인접조사가 이루어 질 수도 있다.

5. 실험 및 분석

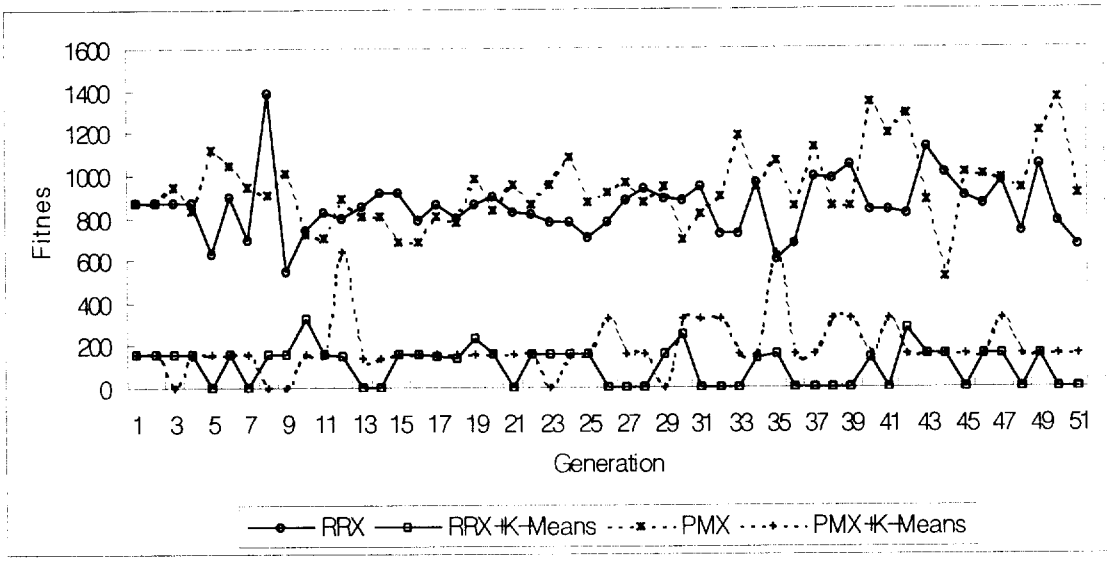
본 연구에서 제안된 방법론을 평가하기 위하여 실험을 수행하였다. 실험 대상 문제는 두 형태로 최적해를 쉽게 구할 수 있는 문제와 제한된 시간 내에 쉽게 최적해를 구할 수 없는 문제를 다루었다. 지역 분할 문제를 풀기 위한 유전자 알고리듬과 K-평균법은 C++로 프로그래밍하여 Windows 95 환경의 pentium 120MHZ의 컴퓨터에서 수행되었다.

5.1 최적해를 쉽게 구할 수 있는 문제

최적해를 구할 수 있는 문제로서는 20×20의 격자에 각 노드의 특성치로 0, 2, 4, 6, 8 중 하나를 할당하였다. 이 때 같은 특성치의 노드들은 서로 인접하도록 하여 연이은 4개 행의 모든 노드들을 같은 값으로 할당하였다. 즉, 분할할 그룹의 수를 5로 하였을 때 최적해는 같은 특성치의 노드들을 같은 그룹으로 묶는 것으로서 목적함수 값은 0이 된다. 유전자 알고리듬을 적용

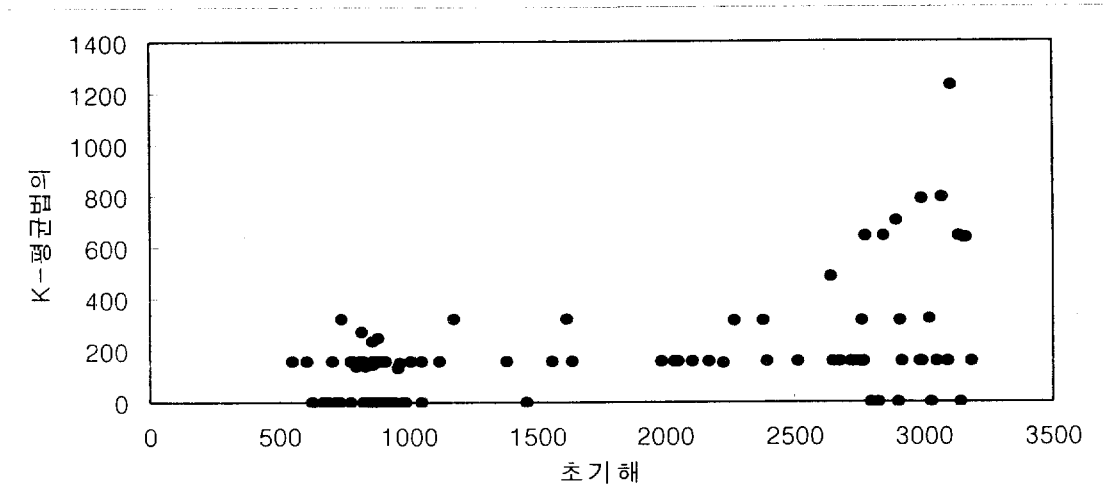
하는데 있어 개체 수 100, 교차확률 0.9, 전위확률 0.6으로 하여 50세대 동안 실험하였다. 교차 연산자로는 PMX와 RRX를 각각 사용하였다. 개체 수 및 필요한 모수값은 예비 실험을 통하여 우수한 성능을 가져온다고 고려되는 것을 선택하였다. [그림 2]는 유전자 알고리듬만을 적용한 경우와 유전자 알고리듬 각 세대에서의 가장 좋은 해에 수정된 K-평균법을 적용시킨 경우에 있어 각 세대의 최소값에 대한 결과를 나타낸다. 후자의 경우 문제에서 특성치의 구조가 좋아 실험 초기에 최적해에 도달하였으나, 유전자 알고리듬 만을 적용한 경우에는 최적해와 거리가 먼 결과를 가져 왔다. 이는 유전자 알고리듬에서 염색체를 해로 복호화하는 방법이 최적해에 유사한 해를 발생 시키지 못하고 있기 때문이다. 실험에 쓰인 두 교차 연산자는 최적해를 생성하는데 는 큰 차이를 보이지 않으나 각 세대의 최소값에 대한 평균과 표준편차에서 약간의 차이를 보인다. 유전자 알고리듬만을 사용하여 RRX를 적용한 경우, 평균과 표준편차는 각각 850.0, 137.8 이고, PMX를 적용한 경우, 각각 932.2, 167.2이다. RRX 연산자는 PMX 연산자에 비해 보다 안정적인 결과를 가져온다. 이는 두 부모 개체로부터 자손 개체를 생성하는데 있어 RRX 연산자가 PMX 연산자보다 부모의 형질을 더 많이 계승 시키는데 기인한다.

K-평균법은 초기해에 따라 서로 다른 결과를 가져온다. [그림 3]은 임의의 100개 염색체에 대한 목적함수 값과 이를 K-평균법에 의해 부분 최적화 시킨 목적함수 값과의 관계를 보여준다. 그림에서 보듯이 초기해의 값과 K-평균값 사이에 비례관계가 있다고 볼 수 없다. 이러한 결과는 유전자 알고리듬에 의한 해를 초기해로 이용하는 대신에 임의로 초기해를 발생시켜 이를 K-평균법에 적용시켜도 좋은 결과를 가져올 수 있음을 시사한다. 그러나, 이를 검증하는 부가적인 실험으로부터 유전자 알고리듬에 의한 해를 K-평균법의 초기해로 사용하는 것이 임의



[그림 2]유전자 알고리듬과 K-평균법에 의한 결과(최적해를 알 수 있는 경우)

의 해를 사용하는 것 보다 좋은 결과를 가져옴을 알 수 있다. 〈표 2〉은 50개의 염색체를 임의로 발생시킨 경우, 이 임의의 염색체를 K-평균법에 적용 시킨 경우, 100개의 개체수로 50세대 동안 유전자 알고리듬에 의한 경우, 그리고, 유전자 알고리듬의 각 세대에서의 최소값을 갖는 염색체에 K-평균법을 적용시킨 경우에 대한 결과를 보여 준다. 유전자 알고리듬에서 교차연산자로서는 RRX를 사용하였고, 필요한 파라미터의 값은 [그림 2]의 실험에 쓰인 값을 사용하였다. 임의로 초기해를 발생하여 K-평균법을 적용시킨 경우에도 최적값에 도달할 수 있으나, 유전자 알고리듬과 K-평균법에 의한 경우에 비교하여 상대적으로 높은 평균과 큰 분산을 가져온다. 반면 유전자 알고리듬과 K-평균법을 사용한 경우에는 최적값을 발생시키는 회수가 많고, 최적값에 근사한 값을 발생시키는 매우 안정적인 결과를 보인다. CPU 시간의 비교에서는 K-평균법의 초기해로 임의의 해나 유전자 알고리듬의 해를 사용하는데 차이가 거의 없다. 100개의 개체를 유전자 알고리듬에 적용하는 부가적인 계산절차에도 불구하고 차이가 없음은 K-평균법의 계산시간에 큰 차이가 있음을 의미한다. 〈표 2〉에서 쉽게 알 수 있듯이 50번의 K-평균법에 대한 계산시간은 임의의 해를 초기해로 하는 경우 835초(한 해당 16.7초), 유전자의 해를 초기해로 하는 경우 381초(한 해당 7.6초)이다. 임의의 해인 경우 보다 유전자에 의한 해가 최



[그림 3]초기해와 K-평균법에 의한 해의 관계

〈표 2〉 각 방법의 비교

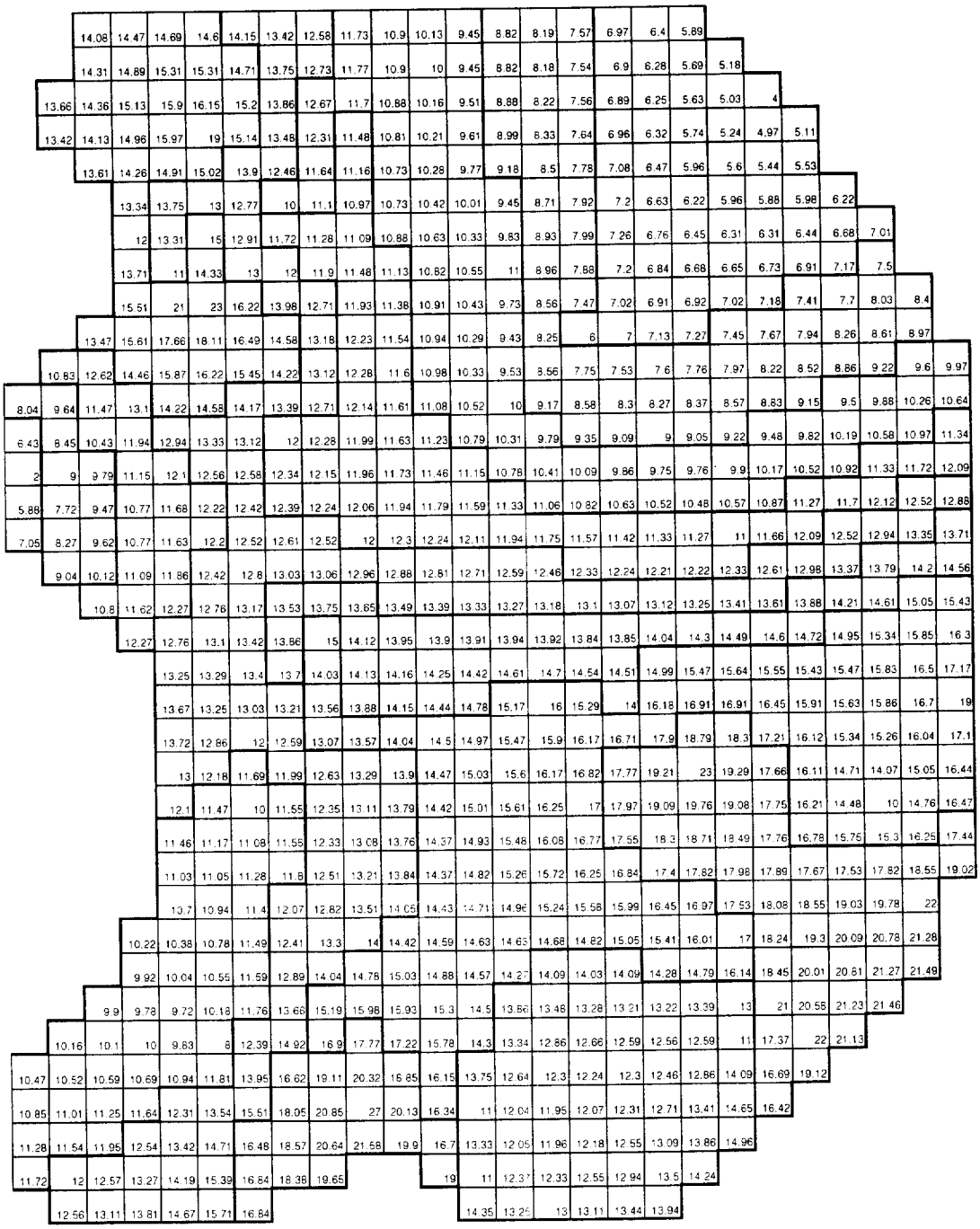
척도 \ 방법	RANDOM	RANDOM+ K-평균법	유전자(RRX)	유전자(RRX)+ K-평균법
평균	2506.7	280.9	835.6	105.7
표준편차	611.6	252.3	136.8	88.7
CPU시간(초)	19	854	463	844

적값에 근사하여 K-평균법의 계산시간이 크게 절약되기 때문이다.

5.2 최적해를 쉽게 구할 수 없는 문제

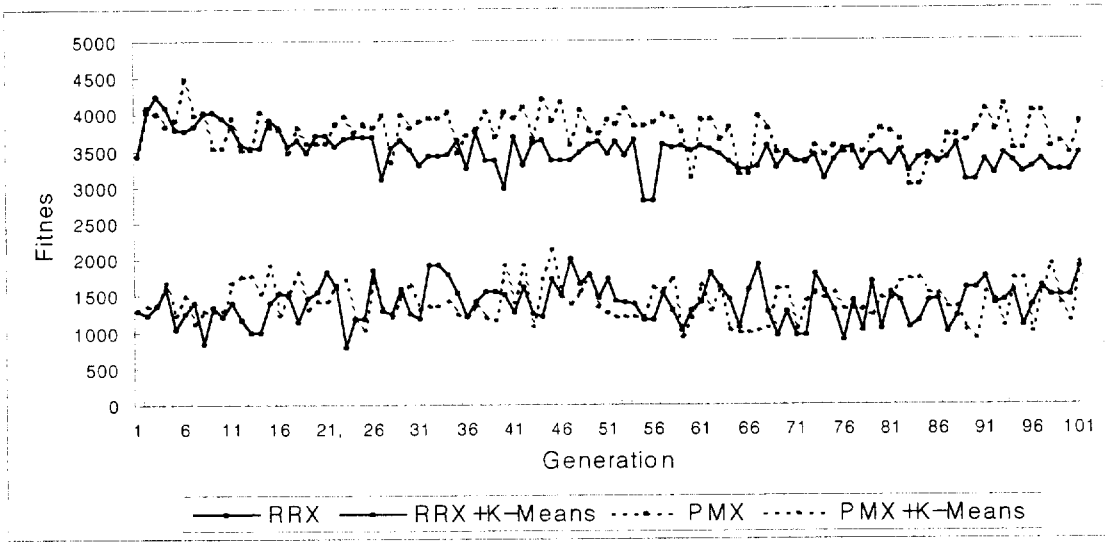
최적해를 쉽게 구할 수 없는 문제로서, 국내를 대상으로 SO_2 대기오염 농도가 서로 상이한 부분 지역으로의 분할을 다룬다. 고려되는 지역에 대한 대기 오염도의 공간적 분포를 정확하게 측정하기 위하여는 대기오염 측정소의 수와 위치에 대한 결정이 매우 중요하다. 측정소의 수는 일반적으로 대기 오염도 측정의 목적, 투자 예산의 한계, 요구되는 오염도 추정의 정확성 등 많은 요인에 의해 결정된다. 이러한 측정소의 수에 대한 결정 문제는 본 연구의 범위를 벗어나므로 실험의 목적에서 제외한다. 고려되는 지역의 오염도에 대한 정확하고, 신뢰성 있는 공간적인 분포의 측정을 위한 목적하에서 측정소 위치를 결정한다면, 그 지역을 오염도가 서로 상이한 부분 지역들로 분할하는 방법론을 사용할 수 있다[8, 9]. 이때, 각 부분 지역내 각 지점에서의 오염도가 유사하도록, 설치될 측정소 수 만큼의 부분 지역으로 분할한다. 각 부분 지역에는 그 지역의 오염도를 대표하도록 하나의 측정소를 위치토록 한다. 한 부분 지역 내의 모든 지점에서 오염도가 유사하다면, 그 지역에 둘 이상의 측정소를 위치토록 할 필요가 없기 때문이다. 분할된 부분 지역 내의 측정소 위치는 지리적 조건, 오염도 대표성 정도 등에 의해 결정될 수 있다. 이러한 부분 지역 내의 측정소 위치 결정 또한 본 연구의 범위를 벗어나므로, 본 실험의 목적에서 제외한다. 본 연구에서 수행하는 지역 분할을 위해서는 지역을 구분한 각 격자에서의 오염도 수치가 있어야 한다. 이를 위해서는 대기확산 모형을 이용하는 방법과 기존의 측정 데이타로서 전체 격자에 대한 Kriging 추정[2]을 하는 방법 등이 있다.

본 실험에서는 전국의 SO_2 대기오염 농도 분포를 정확하게 측정할 목적으로, 15개의 측정소를 위치시킬 부분 지역으로 분할 한다고 가정한다. 가로 30, 세로 40의 격자로 전국을 분리하여, 기 설치된 측정소에서의 96년도 연평균 SO_2 측정값을 입력으로 Kriging 추정을 수행하였다. [그림 4]는 Kriging 추정한 결과로서 불필요한 격자를 제외한 각 격자에서의 SO_2 대기 농도를 나타낸다. 유전자 알고리듬에 필요한 모수 값은 예비 실험을 통하여 전체 개체 수를 200, 세대 수를 100, 교차 확률을 0.9, 전위 확률을 0.6으로 정하였다. K-평균법은 각 세대에서의 최소값을 갖는 해에만 적용하였다. [그림 5]은 유전자 알고리듬만을 적용한 경우와 부가적으로 K-평균법을 적용한 경우에 있어 교차 연산자로서 각각 PMX와 RRX를 사용한 결과를 보여준다. 최적해를 아는 경우의 실험 결과와 같이, 유전자 알고리듬만을 적용한 경우와 부가적으로 K-평균법을 적용한 경우 사이에 현저한 차이를 보인다. K-평균법을 부가적으로 사용한 경우가 우월한 해를 발생시킨다. [표 3]에서 나타나듯이 4가지 경우 중 교차 연산자를 RRX로 하여 K-평균법을 사용한 경우가 가장 좋은 결과를 가져온다. 100 세대의 결과에 대한 평균치에서 뿐만 아니라, 전 세대의 최소값에서 805.9로 가장 우월한 해를 발생시킨다. [그림 4]에서 굵은 선으로 구분된 부분 지역들이 가장 좋은 해의 지역 분할을 나타낸다.



[그림 4] Kriging 추정에 의한 SO_2 대기 농도(단위 : 10^{-3} PPM)

[그림 5] 유전자 알고리듬과 K-평균법에 의한 결과



〈표 3〉 최적해를 알 수 없는 경우에서의 유전자 알고리듬 비교

척도 \ 방법	유전자(RRX) 알고리듬	RANDOM+ K-평균법	유전자(PMX) 알고리듬	유전자(RRX)+ K-평균법
평균	3480.2	1392.8	3738.8	1419.6
표준편차	252.7	267.6	270.6	261.4
최소값	2802.3	805.9	3020.1	907.4

7. 결 론

본 연구에서는 지역 분할 문제를 해결하기 위한 방법론으로서 유전자 알고리듬과 수정된 K-평균법을 제시하고, 제안된 방법론의 성능 분석을 위한 실험을 수행하였다.

주어진 지역을 부분지역으로 분할하는 문제는 제한된 시간 내에 최적해를 구하기가 쉽지 않다. 때문에, 최적해는 아니지만 제한된 시간 내에 근사해를 구하는 것이 보다 의미가 있다. 유전자 알고리듬은 이러한 문제를 해결하는 우수한 방법으로 알려져 있다. 그러나, 지역분할 문제에 적용키 위하여는 새로운 유전자 암호화와 이에 알맞은 유전 연산자를 고안하는 것이 중요하다. 본 연구에서 지역분할 문제에 적용할 수 있는 edge 우선순위 표현방법을 제안하였다. 이 표현방법은 각 노드들이 속한 그룹을 구하여야 하는 추가적인 복호화 절차가 필요하나, 전형적인 교차, 전위의 연산자를 사용할 수 있는

장점이 있다. 교차 연산자로는 판매원 문제를 풀기위한 유전자 알고리듬에서의 PMX등과 본 논문에서 제안하는 RRX를 사용할 수 있다. 유전자 알고리듬에 의한 해는 유전자의 복호화 방법에 대한 제약으로 인하여 최적해를 발생시키기가 매우 어렵다. 유전자 알고리듬에 의한 해를 보다 좋은 해로 변형시키는 방법으로서 개체 분할 문제에 사용하는 K-평균 방법을 수정하여 이용하였다. 즉, 유전자 알고리듬에서의 한 개체를 해로 복호화한 후 이를 초기해로 하여 K-평균 방법으로 해를 향상시킬 수 있다. K-평균 방법은 계산의 복잡성으로 인하여 많은 계산시간을 필요로 한다. 실험결과는 RRX를 교차 연산자로 한 유전자 알고리듬 각 세대에서의 해에 K-평균 방법을 적용하는 것이 우수한 해를 빠른 시간 내에 발생시킬 수 있음을 보여준다.

참 고 문 헌

[1] Bhuyan, J.N., Raghavan, V.V., and Elayavalli, V.K., "Genetic Algorithm for Clustering with an Ordered Representation," in Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann Publishers, Los Altos, CA, 1991, pp.408-415.

[2] Cressie, N.A.C. , *Statistics for Spatial Data*, John Wiley & Sons, New York: NY, 1991.

[3] Davis, L., "Applying Adaptive Algorithms to Epistatic Domains," in Proceedings of the International Joint Conference on Artificial Intelligence, 1985, pp.162-164.

[4] Fisher, W.D., "On Grouping for Maximum Homogeneity," Journal of American Stat. Assoc., Vol.53(1958), pp.789-798.

[5] Goldberg, D.E. and Lingle, R., "Allelis, Loci, and the TSP", in Proceedings of the First International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, NJ, 1985, pp.154-159.

[6] Holland, J., *Adaptation in Neural and Artificial Systems*, Univ. of Michigan Press, 1988.

[7] Jones, D.R. and Bertramo, M.A., "Solving Partitioning Problems with Genetic Algorithms", in Proceedings of the Fourth International Conference on Genetic Algorithms, Morgan Kaufmann Publishers, Los Altos, CA, 1991, pp.442-449.

[8] Nakamori, Y., Ikeda, S.and Sawaragi, Y., "Design of Air Pollutant Monitoring System by Spatial Sample Stratification", *Atmospheric Environment*, Vol. 13(1979), pp.97-103.

[9] Nakamori, Y., "Interactive Design of Urban Level Air Quality Monitoring Network", *Atmospheric Environment*, Vol. 18, No. 4(1984), pp.793-799.

[10] Oliver, I.M., Smith, D.J., and Holland, J.R.C., "A Study of Permutation Crossover Operators on the Traveling Salesman Problem", in Proceedings of the Second International Conference on Genetic Algorithms, Lawrence Erlbaum Associates, Hillsdale, NJ, 1987, pp.224-230.

[11] Poon, P.W. and Carter, J.N., "Genetic Algorithm Crossover Operators for Ordering Applications", *Computers & Operations Research*, Vol. 22, No. 1(1995), pp.135-147.

[12] Ward, J., "Hierarchical Grouping to Optimize an Objective Function", *Journal of the American Statistical Association*, Vol. 58(1963), pp.236-244.