

서열순서화문제를 위한 상위정보를 이용하는 혼합형 유전알고리즘

A Hybrid Genetic Algorithm Using Epistasis Information for Sequential Ordering Problems

서동일* · 문병로**

Dong-Il Seo and Byung-Ro Moon

* 서울대학교 컴퓨터공학부

** 서울대학교 컴퓨터공학부

요 약

본 논문에서는 서열순서화문제를 위한 새로운 혼합형 유전알고리즘을 제안한다. 제안된 유전알고리즘에서는 보로노이양자화교차를 교차연산자로 사용하고 경로보전 3-최적화를 지역탐색 휴리스틱으로 사용한다. 보로노이양자화교차는 주어진 문제 인스턴스의 상위 정보를 이용하는 교차연산자이다. 이것은 원래 순회판매원문제를 위해서 제안된 교차연산자이기 때문에 서열순서화문제에 적용하기 위해서는 상당한 변형을 필요로 한다. 본 연구에서는 서열순서화문제에 맞도록 보로노이양자화교차를 적절히 변형하고, 변형된 보로노이양자화교차에서 필요로 하는 가능해생성알고리즘, 선행관계사이클분해알고리즘, 유전자거리지정방법 등을 개발하였다. TSPLIB와 ZIB-MP-Testdata로부터 얻어진 서열순서화문제 인스턴스들에 대한 실험결과, 제안된 유전알고리즘이 비교된 다른 유전알고리즘들에 비해서 더 안정적이고 성능이 우수한 것으로 나타났다.

Abstract

In this paper, we propose a new hybrid genetic algorithm for sequential ordering problem (SOP). In the proposed genetic algorithm, the Voronoi quantized crossover (VQX) is used as a crossover operator and the path-preserving 3-Opt is used as a local search heuristic. VQX is a crossover operator that uses the epistasis information of given problem instance. Since it is a crossover proposed originally for the traveling salesman problem (TSP), its application to SOP requires considerable modification. In this study, we appropriately modify VQX for SOP, and develop three algorithms, required in the modified VQX, named Feasible Solution Generation Algorithm, Precedence Cycle Decomposition Algorithm, and Genic Distance Assignment Method. The results of the tests on SOP instances obtained from TSPLIB and ZIB-MP-Testdata show that the proposed genetic algorithm outperforms other genetic algorithms in stability and solution quality.

Key words : 서열순서화문제, 혼합형 유전알고리즘, 상위, 링크지, 보로노이양자화교차

1. 서 론

정의된다.

n 개의 노드들로 이루어진 집합 $V = \{1, 2, \dots, n\}$ 가 주어졌을 때 이 노드들의 일부 또는 전체를 중복 없이 임의의 순서로 방문하는 것을 경로라고 한다. 어떠한 경로가 전체 노드를 포함할 때 이를 해밀턴경로라고 한다. 주어진 노드들에 대한 해밀턴경로에서, 가령 노드 i 는 항상 노드 j 의 앞에 오도록 하는 조건을 생각할 수 있는데, 이러한 조건을 선행관계조건이라고 하고 $i < j$ 로 표시한다. 이 때, 노드 i 를 노드 j 의 선행노드라고 하고, 반대로 노드 j 를 노드 i 의 계승노드라고 한다. 임의의 노드 i 에서 임의의 노드 j 로 가는데 필요한 비용 $c_{ij} \geq 0$ 가 주어질 때 이러한 비용들을 모아 놓은 행렬 $C = (c_{ij})$ 를 비용행렬이라고 한다. 따라서 해밀턴경로 $\pi = (\pi_1, \pi_2, \dots, \pi_n)$, $\pi_i \in V$ 의 비용 $C(\pi)$ 는 다음과 같이

$$C(\pi) = \sum_{i=1}^{n-1} c_{\pi_i, \pi_{i+1}} \quad (1)$$

서열순서화문제는 주어진 선행관계조건을 만족하는 최소 비용의 해밀턴경로를 찾는 문제로 Escudero[1]에 의해서 처음으로 소개되었다. 일반적으로, 시작 노드는 노드 1로, 끝 노드는 노드 n 으로 고정된다. 즉, 선행관계조건 $1 < i < n$, $i \in V - \{1, n\}$ 이 묵시적으로 주어진다고 가정한다. 일반적인 비용행렬은 비대칭이고 선행관계조건들은 추이적이며 비순환적이다. 서열순서화문제를 종종 “선행관계조건을 가지는 비대칭 해밀턴경로문제”라고 부르기도 한다. 선행관계조건을 갖지 않는 서열순서화문제는 선행관계조건을 갖는 서열순서화문제의 특수한 경우로써 자연스럽게 순회판매원문제로 축소된다. 순회판매원문제가 NP-hard 문제[2]이므로 서열순서화문제도 NP-hard 문제이다.

최적성을 보장해줄 수는 없지만 주어진 시간 내에 품질이 우수한 해를 효율적으로 찾아내는데 초점을 맞춘 접근방법들이 최근 들어 집중적으로 연구되기 시작했는데, 이러한 방법

접수일자 : 2005년 9월 14일
완료일자 : 2005년 11월 16일

들을 보통 휴리스틱이라고 부르고, 이들 중 폭넓은 적용범위를 가지는 방법들을 메타휴리스틱이라고 부른다. 최근 들어 알려진 사실은 메타휴리스틱을 독립적으로 사용하는 것보다 지역탐색휴리스틱과 결합하여 사용하는 것이 훨씬 강력하다는 것이다. 이러한 형태의 휴리스틱을 혼합형 메타휴리스틱이라고 부른다. 서열순서화문제를 풀기 위한 지역탐색휴리스틱으로는 경로보전 3-최적화와 서열순서화문제-3-교환 등이 있다[3]. 유전알고리즘은 메타휴리스틱의 대표적인 예로서 집단유전학에 기초를 둔 문제해결기법이다. Chen과 Smith [4]가 제안한 유전알고리즘에서는 최대부분순서/임의삽입이라고 하는 교차연산자가 사용된다. 이 외에도 서열순서화문제를 대상으로 하는 유전알고리즘에 대한 여러 연구가 진행되었다[5,6].

본 논문에서는 서열순서화문제를 풀기 위한 새로운 혼합형 유전알고리즘을 제안한다. 이를 위해서 이미 순회판매원 문제에 성공적으로 적용된 바 있는 보로노이양자화교차[7]를 사용한다. 이 교차연산자를 서열순서화문제에 적용하기 위해서는 상당한 변형과 여러 가지 부수적인 알고리즘들을 필요로 한다. 본 논문에서는 서열순서화문제에 맞도록 보로노이양자화교차를 적절히 변형하고, 이 때 필요한 가능해생성 알고리즘, 선행관계사이클분해알고리즘, 유전자거리지정방법 등을 새롭게 제안한다.

본 논문의 구성은 다음과 같다. 2절에서 배경을 설명하고 3절에서 제안된 유전알고리즘을 설명한다. 4절에서 실험결과를 설명하고 5절에서 결론을 맺는다.

2. 배 경

최적화문제는 일반적으로 (U, f) 로 부호화되는데 모집단이라고 부르는 U 는 가능해들의 집합을 의미하고 적합도함수라고 부르는 f 는 U 에서 실수집합으로 가는 함수가 된다. 적합도함수는 주어진 해가 얼마나 적합한 해인지를 수치화한 것이다. 각각의 해는 보통 하나 이상의 변수들로 표현되는데, 각 변수들의 정의역이 이산적인 경우 조합최적화문제라고 부른다. 최적화문제의 변수들을 유전알고리즘에서는 특히 유전자라고 부르고 해의 유전자 전체를 염색체라고 부른다. 조합 최적화문제에서 적합도에 대한 어떤 변수의 기여도(또는 영향)가 다른 변수들의 값에 의존하는 현상이 관찰할 수 있는데, 이러한 현상을 상위 또는 링키지라고 부른다[8]. 상위는 많은 최적화문제들을 최대경사법과 같은 단순접근법을 이용해서 쉽게 풀 수 없도록 하는 주요한 원인으로 작용한다. 최근까지 진화알고리즘 분야에서의 많은 연구가 상위와 관련되어 있다.

유전알고리즘[9,10]은 주어진 문제의 해들 각각을 개체로 보고 해들을 모아 놓은 해집단을 집단적으로 진화시켜서 원하는 해를 찾아나가는 문제해결기법이다. 이를 위해서 일반적으로 선택연산자, 교차연산자, 변이연산자, 대치연산자 등이 사용된다. 혼합형 유전알고리즘은 보통 선택연산자, 교차연산자 및 변이연산자를 통해서 만들어진 해를 해집단 내에 대치하기 전에 지역탐색휴리스틱을 이용해서 지역최적해로 변환한다.

교차연산 과정에서 유전자들 간의 상위를 반영하기 위한 방법들 중 하나는 각각의 유전자들을 유클리드공간 또는 비유클리드공간 내의 임의의 위치에 대응시키는 방법이다. 유전자들이 대응되는 가상공간을 염색체공간이라고 부르고 그 공간의 구조를 염색체구조라고 부른다. 전통적인 염색체구

조는 일차원 배열이다. 일반적으로 유전자들 간의 상위는 비선형적인 복잡한 구조를 가지므로, 일차원 배열과 같은 단순한 염색체구조로는 이러한 복잡한 구조를 잘 표현하기 어렵다. 따라서 보다 고차원적인 염색체구조를 생각할 필요가 있는데, 이러한 염색체구조로는 이차원 배열, 이차원 실공간(즉, 평면), 완전그래프 등이 있다[11]. 보로노이양자화교차[7]는 완전그래프를 염색체구조로 사용하는 교차연산자이다. 다음의 절들에서는 접근법과 연구결과들에 대해서 설명한다.

3. 제안된 유전알고리즘

3.1 부호화

주어진 최적화문제를 유전알고리즘을 이용하여 풀기 이전에 먼저 부호화방법을 정해야 한다. 일반적으로 해를 표현하기 위한 변수들과 그 해의 적합도함수를 정해주게 된다. 본 논문에서는 서열순서화문제를 부호화하기 위해서 Bui와 Moon[12]이 순회판매원문제에 적용한 바 있는 위치기반부호화를 사용한다. 즉, n 개의 노드들에 대해서 n 개의 유전자를 하나씩 순서대로 할당하고 n 번째 유전자를 제외한 각 유전자는 주어진 해밀턴경로 상에서 대응되는 노드의 다음 노드의 인덱스에 해당하는 값을 저장한다. 마지막 n 번째 유전자는 경로의 끝을 나타내기 위해서 -1 을 저장한다. 해의 적합도로는 식 (1)에 의해서 정의되는 해밀턴경로의 비용을 그대로 사용한다.

3.2 보로노이양자화교차

3.2.1 보로노이양자화교차의 구조

보로노이양자화교차[7]는 염색체구조가 가중치완전그래프(weighted complete graph)인 교차연산자이다. 즉, 유전자들로 이루어진 완전그래프 상에서 교차연산이 이루어진다. 그래프의 가중치가 대칭적이면 비방향그래프가 사용되고 그렇지 않으면 방향그래프가 사용된다. 이 때, 유전자들 간의 가중치는 대응되는 유전자들 간의 상위의 강도에 반비례하는 값으로 정해진다. 이러한 가중치를 보로노이양자화교차에서는 유전자거리라고 부른다. 이러한 염색체구조를 사용함으로써 비선형적인 복잡한 상위 구조를 교차에 반영시킬 수는 장점이 있다. 본 논문에서는 유전자거리를 휴리스틱한 방법으로 지정하는데 이 지정방법은 3.3절에서 설명한다.

보로노이양자화교차는 단순한 구조를 가지는데, 그림 1은 그 의사코드를 보여준다. n 개의 유전자들로 구성된 완전그래프 염색체구조에서 각 유전자들 사이의 유전자거리가 $D=(d_{ij})$ 가 주어졌다고 하자. 주어진 유전자거리에 의해서 정의되는 염색체공간이 k 개의 보로노이영역 l 으로 분할된다.

1) l 차원 공간상에서 k 개의 중심점이 주어질 때 한 중심점에 대응되는 보로노이영역은 그 중심점이 가장 가까운 중심점이 되는 점들의 집합을 의미한다. 즉, 중심점의 주변영역을 의미한다[13]. 예를 들어, k 개의 중심점 $v_1, v_2, \dots, v_k \in R^l$ 가 주어졌을 때 v_i 에 대응되는 보로노이영역은 $\{x \in R^l : |x - v_i| < |x - v_j|, \forall j = 1, 2, \dots, k, j \neq i\}$ 이 된다.

1. $VQX(n, k, D, p_1, p_2)$
2. {
3. n 개의 유전자들 중에서 k 개의 중심유전자를 임의로 골라 주어진 유전자거리 $D=(d_{ij})$ 에 의해서 만들어지는 염색체공간을 k 개의 보로노이영역으로 분할한다;
4. 각 중심유전자의 색깔을 **백색** 혹은 **회색**으로 임의로 지정한다;
5. 각 유전자 i 에 대해서 다음을 수행한다 {
6. 유전자 i 가 속한 보로노이영역의 중심유전자의 색깔이 **백색**이면 부모해 p_1 의 유전자 값을 자식해 o 로 복사하고, **회색**이면 부모해 p_2 의 유전자 값을 o 로 복사한다;
7. }
8. o 를 반환한다;
9. }

그림 1. 보로노이양자화교차.

Fig. 1. Voronoi quantized crossover.

이때 각 보로노이영역의 중심의 역할을 하는 k 개의 유전자들이 임의로 선택되는데 이들을 중심유전자라고 부른다(그림 1의 3행). 각 보로노이영역의 색깔을 백색 혹은 회색으로 임의로 지정한다(4행). 이제 자식해 o 의 각 유전자에 대해서 그 유전자가 속한 보로노이영역의 색깔이 백색이면 부모해 p_1 으로부터 유전자 값을 복사하고, 회색이면 부모해 p_2 로부터 복사한다(5-7행). 마지막으로 자식해 o 를 반환한다(8행). 보로노이양자화교차는 순회판매원문제에 처음으로 적용되었는데, 이를 서열순서화문제에 적용하기 위해서는 상당한 변형을 필요로 한다.

3.2.2 서열순서화문제를 위한 보로노이양자화교차

그림 2는 서열순서화문제에 맞도록 변형된 보로노이양자화교차의 의사코드를 보여준다. 그림 2의 1-7행은 그림 1의 1-4행과 같다. 즉, 그림 2의 교차가 그림 1의 교차와 다른 점은 8-12행이 된다. 3.1절에서 설명한 위치기반후호화에 의해서 각 유전자의 값은 해밀턴경로 상에서 그 유전자에 대응되는 노드의 다음 노드를 가리킨다. 따라서 각 유전자의 값은 그 유전자에 대응되는 노드에서 출발하는 방향간선을 의미하게 된다. 부모해 p_1 에 대응되는 해밀턴경로의 각 방향간선에 대해서 그것의 양 끝 노드가 모두 색깔이 **백색**인 보로노이영역에 속하면 그 값을 자식해 o 로 복사한다(9행). 마찬가지로 부모해 p_2 에 대응되는 해밀턴경로의 각 방향간선에 대해서 그것의 양 끝 노드가 모두 색깔이 **회색**인 보로노이영역에 속하면 그 값을 자식해 o 로 복사한다(10행). 이러한 과정을 거치는 이유는 양 부모해의 유전적인 성질을 자식해가 최대한 물려받으면서도 자식해를 가능해로 만드는 것이 어렵지 않도록 하기 위해서이다. 이러한 과정을 통해서 생성된 자식해 o 는 각 방향간선의 양 끝점이 모두가 같은 색깔의 보로노이영역에 속했을 때에만 부모로부터 물려받았으므로 경로단편들로 이루어진 불완전한 해가 된다. 이 해를 가능해로 만들기 위해서 프로시저 **GenerateFeasibleSolution()**을 호출한 다음, 만들어진 가능해를 반환한다(12-13행). 그림 2의 6행에서 유전자거리가 같은 경우가 발생하면 해당 중심유전자들 중에서 랜덤하게 선택한다.

1. $VQX-SOP(n, k, D, p_1, p_2)$
2. {
3. 유전자의 인덱스 집합을 $I←\{1, 2, \dots, n\}$ 라고 하고 중심유전자의 인덱스 집합을 $K←\{1, 2, \dots, k\}$ 라고 두자;
4. k 개의 중심유전자 $\{s_1, s_2, \dots, s_k\} \subset I$ 를 임의로 고른다;
5. 중심유전자의 색깔 $c_K[i], i \in K$ 를 **백색**과 **회색** 중에서 하나로 임의로 지정한다;
6. 각 유전자 i 에 대해서 그 유전자가 속하는 보로노이영역 $\mathcal{N}[i], i \in I$ 를 그 유전자까지의 유전자거리가 가장 가까운 중심유전자의 인덱스로 지정한다. 즉, 주어진 유전자거리 $D=(d_{ij})$ 에 대해서 $\mathcal{N}[i]←\arg \min_{j \in K} \{d_{s_j, i}\}$;
7. 각 유전자의 색깔 $c_I[i], i \in I$ 를 그 유전자의 보로노이영역 $\mathcal{N}[i]$ 의 중심유전자의 색깔 $c_K[\mathcal{N}[i]]$ 로 지정한다. 즉, $c_I[i]←c_K[\mathcal{N}[i]]$;
8. 각 유전자 i 에 대해서 다음을 수행한다 {
9. 만약 유전자 i 의 색깔 $c_I[i]$ 와 유전자 $p_1[i]$ 의 색깔 $c_I[p_1[i]]$ 가 모두 **백색**이면 자식해 o 의 i 번째 유전자 $o[i]$ 를 $p_1[i]$ 로 지정한다;
10. 만약 유전자 i 의 색깔 $c_I[i]$ 와 유전자 $p_2[i]$ 의 색깔 $c_I[p_2[i]]$ 가 모두 **회색**이면 자식해 o 의 i 번째 유전자 $o[i]$ 를 $p_2[i]$ 로 지정한다;
11. }
12. o 를 가능해로 만든다. 즉, $o←\text{GenerateFeasibleSolution}(o)$;
13. o 를 반환한다;
14. }

그림 2. 서열순서화문제를 위한 보로노이양자화교차.

Fig. 2. Voronoi quantized crossover for sequential ordering problem.

3.2.3 가능해생성알고리즘

그림 3은 경로단편들로 이루어진 자식해 o 를 가능해로 만들기 위한 가능해생성알고리즘의 의사코드를 보여준다. 그림 2의 3-11행에서 만들어진 자식해 o 는 경로단편들로 구성되는데 각 경로단편들은 방향을 가진다. 이러한 경로단편들의 집합을 S 라고 하자(그림 3의 3행). S 내의 경로단편들은 두 부모로부터 물려받았기 때문에 선행관계조건들과 함께 보면 사이클을 이룰 수 있다. 따라서 우선적으로 이러한 사이클을 제거해주어야 하는데, 이것은 사이클을 이루는 임의의 한 경로단편을 더 작은 단편들로 나눔으로써 해결할 수 있다. 이 과정은 프로시저 **DecomposePrecedenceCycle()**을 호출함으로써 이루어진다(4행). 이제, 노드 1을 포함하는 경로단편에서부터 시작하여 경로단편들을 하나씩 붙여서 하나의 해밀턴경로로 만드는 과정을 거치는데, 이 과정은 다음과 같이 탐리적으로 이루어진다. 먼저 노드 1을 포함하는 경로단편을 s_0 라고 두고 이를 S 에서 제거한다(5행). 그런 다음, S 에서 s_0 뒤에 붙일 수 있는, 즉 선행관계조건을 위반하지 않는 경로단편들 중에서 s_0 의 끝 노드로부터 그 경로단편의 시작 노드까지의 비용이 가장 작은 것을 하나 골라서 s_0 의 뒤에 붙이기를 반복한다(6-9행). 이렇게 함으로써 선행관계조건을

위반하지 않으면서 그 비용이 비교적 적은 해밀턴경로를 만들 수 있게 된다. 이렇게 생성된 해밀턴경로를 나타내는 가능해 o' 을 만들고 이를 반환한다(10-11행).

```

1. GenerateFeasibleSolution (o)
2.
3.   해 o의 모든 경로단편들의 집합을 S라고 두자;
4.   S의 경로단편들이 선행관계조건들과 함께 사이클을 이룰 경우 이 사이클을 제거한다. 즉,
      S ← DecomposePrecedenceCycle (S);
5.   S에서 노드 1을 포함하는 경로단편을 s0라고 두고 이를 S로부터 제거한다. 즉, S ← S - {s0};
6.   S가 공집합이 아니면 다음을 반복한다 {
7.     S에서 s0 뒤에 붙일 수 있는(선행관계조건을 위반하지 않는) 경로단편들 중 s0의 끝 노드로부터 그 경로단편의 시작 노드까지의 비용이 가장 적은 것을 s라고 두고 이를 S로부터 제거한다. 즉, S ← S - {s};
8.     s0의 뒤에 s를 붙여서 새로운 s0를 만든다;
9.   }
10.  이제 해밀턴경로 s0에 대응되는 해 o'을 생성한다;
11.  o'을 반환한다;
12. }
    
```

그림 3. 가능해생성알고리즘.

Fig. 3. Feasible solution generation algorithm.

3.2.4 선행관계사이클분해알고리즘

그림 4는 선행관계사이클분해알고리즘의 의사코드를 보여준다. 우선 경로단편 집합 F 와 경로단편들의 선행관계 집합 T 를 각각 공집합으로 둔다(그림 4의 4행). F 는 이미 점검이 끝난 경로단편들을 저장하기 위한 것이다. 모든 경로단편들에 대해서 점검을 마칠 때까지, 즉 F 와 S 가 같아질 때까지 다음을 수행한다(5-21행). 아직까지 점검하지 않은 경로단편들 중에서 하나의 경로단편 s 를 임의로 고른다(6행). 경로단편 s 내의 각 노드 i 와 노드 i 의 모든 선행노드 j 에 대해서 다음을 수행한다(7-8행). 노드 j 를 포함하는 S 내의 경로단편을 p 라고 하자(9행). p 와 s 가 서로 다른 경로단편이고 선행관계조건 (p, s) 가 아직 T 의 원소가 아니면 선행관계조건 (p, s) 를 T 에 넣고 T 가 추이적 닫힘이 되도록 한다(11-12행). 즉, 모든 $(q, p) \in T$ 와 $(s, t) \in T$ 에 대해서 (q, s) , (q, t) 및 (p, t) 를 T 에 넣는다. 이러한 수행과정 중에 반대방향의 선행관계, 즉 (s, p) , (s, q) , (t, q) 또는 (t, p) 가 이미 T 에 들어 있는지를 점검하게 된다. 만일 이미 들어 있다면 이것은 T 내에 사이클이 존재한다는 것을 의미하므로 이것을 분해해야 한다(13-17행). 이 사이클은 경로단편 s 를 두 개의 부분단편 s' 과 s'' 으로 분리함으로써 분해할 수 있다(14행). 분리하는 기준점으로는 노드 i 의 직전위치와 직후위치 중의 하나가 선택되는데, 둘 중에서 결과적으로 만들어지는 두 개의 부분단편의 크기가 서로 균형을 이루는 쪽을 선택한다. 사이클을 분해한 다음, S 내의 s 를 s' 과 s'' 으로 대체한 다음 처음부터 다시 시작한다(15-16행). 이러한 과정은 모든

경로단편에 대해서 이 과정을 수행하더라도 사이클이 발견되지 않을 때까지 반복한다. 이렇게 모든 사이클이 분해한 다음 경로단편들의 집합 S 를 반환한다(22행).

```

1. DecomposePrecedenceCycle (S)
2. {
3.   START:
4.   경로단편집합 F를 F ← ∅로 두고 경로단편들의 선행관계 집합 T를 T ← ∅라고 둔다;
5.   |F| < |S|인 동안 다음을 반복 수행한다 {
6.     S - F로부터 하나의 경로단편 s를 임의로 선택하여 F에 넣는다. 즉, F ← F ∪ {s};
7.     s 내의 각 노드 i에 대해서 다음을 수행한다 {
8.       j < i인 각 노드 j에 대해서 다음을 수행한다 {
9.         노드 j가 속해있는 S 내의 경로단편을 p라고 하자;
10.        만약 p ≠ s이고 선행관계 (p, s)가 이미 T에 들어있지 않으면 {
11.          선행관계 (p, s)를 T에 넣는다. 즉, T ← T ∪ {(p, s)};
12.          T가 추이적 닫힘이 되도록 한다. 즉, 모든 (q, p) ∈ T, (s, t) ∈ T에 대해서 (q, s), (q, t) 및 (p, t)를 T에 넣는다;
13.          만약 12행에서 사이클을 발견하였다면 {
14.            s를 s'과 s''으로 분리하여 사이클을 분해한다;
15.            S에 s 대신 s'과 s''을 넣는다. 즉, S ← S - {s} ∪ {s', s''};
16.            START로 이동한다;
17.          }
18.        }
19.      }
20.    }
21.  }
22.  S를 반환한다;
23. }
    
```

그림 4. 선행관계사이클분해알고리즘.

Fig. 4. Precedence cycle decomposition algorithm.

3.2.5 보로노이양자화교차의 예

그림 5는 서열순서화문제를 위한 보로노이양자화교차의 한 예를 보여준다. 그림에서 노드는 작은 원으로, 명시적으로 주어지지 않는 선행관계조건들은 점선 화살표로 표시하였다. 그림으로 표현할 수 있도록 하기 위해서 유전자거리 $D = (d_{ij})$ 에 의해서 만들어지는 염색체공간이 이차원 평면이라고 가정하였다. (a)는 21개의 노드와 6개의 선행관계조건으로 이루어진 서열순서화문제를 그림으로 표현한 것이다. 각 노드에 하나의 유전자가 대응되므로 각 노드의 위치는 염색체공간에서의 유전자의 위치에 해당한다. (b)는 임의로 고른 9개의 중심유전자에 의해서 염색체공간이 9개의 보로노이영역으로 분할된 것을 보여준다. 각각의 영역에 색깔을 백색 혹은 회색으로 임의로 지정된다. (c)와 (d)는 주어진 두

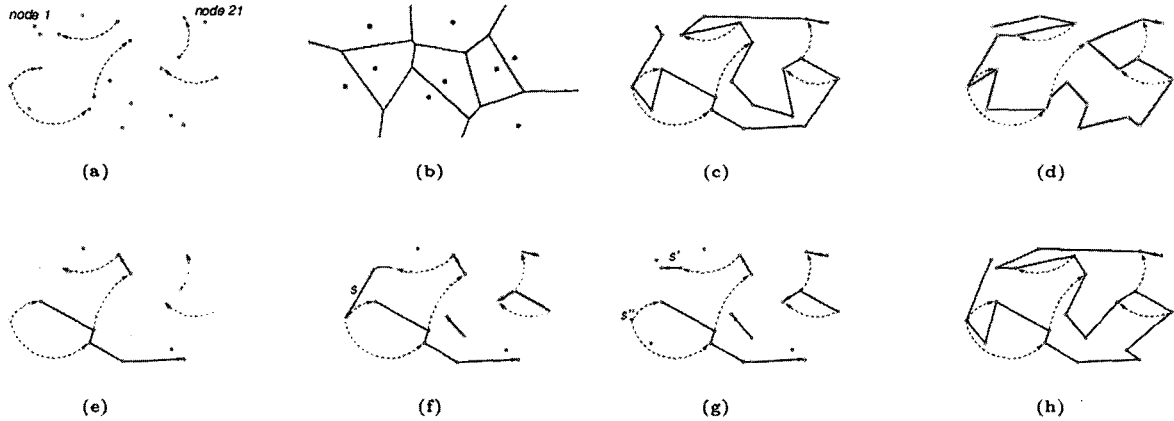


그림 5. 서열순서화문제를 위한 보로노이양자화교차의 한 예. (a) 21개의 노드(유전자)와 6개의 선행관계조건으로 이루어진 서열순서화문제 인스턴스. (b) 임의로 고른 9개의 중심유전자(노드)에 의해서 만들어지는 보로노이영역들의 색깔을 백색 혹은 회색으로 임의로 지정한 결과. (c) 부모해 p_1 . (d) 부모해 p_2 . (e) 부모해 p_1 으로부터 물려받은 경로단편들. (f) 부모해 p_1 과 p_2 로부터 물려받은 경로단편들. (g) 선행관계사이클을 분해한 결과. (h) 탐리적인 방법으로 가능해로 만든 결과.

Fig. 5. An example of VQX for SOP. (a) An SOP instance with 21 nodes (genes) and 6 precedence constraints. (b) After coloring with white or gray color the Voronoi regions determined by randomly chosen 9 center genes (nodes). (c) Parent p_1 . (d) Parent p_2 . (e) Path segments inherited from parent p_1 . (f) Path segments inherited from parents p_1 and p_2 . (g) After decomposing the precedence cycle. (h) Feasible solution constructed by a greedy method.

부모해 p_1 과 p_2 를 각각 보여준다. 두 부모해는 모두 선행관계조건을 잘 만족시킨다. (e)는 자식해 o 가 부모해 p_1 으로부터 물려받은 경로단편들을 보여주고, (f)는 두 부모해 p_1 과 p_2 로부터 물려받은 경로단편들을 함께 보여준다. 이 때, 경로단편들 중 일부가 선행관계조건에 의해서 사이클을 이룰 수 있는데, 이 예에서는 하나의 사이클을 포함하고 있다. (g)는 사이클을 이루는 경로단편들 중 하나인 s 를 두 부분단편 s' 과 s'' 으로 분리한 결과를 보여준다. 마지막으로, (h)는 사이클이 제거된 경로단편들을 탐리적인 방법을 통해 가능해로 만든 결과를 보여준다.

3.3 유전자거리지정방법

유전자거리의 이상적인 값은 상위의 강도에 반비례하는 값이라고 할 수 있다. 그러나 상위의 정확한 값을 얻을 수 있는 실용적인 방법이 아직까지 알려져 있지 않기 때문에, 본 논문에서는 휴리스틱에 의존하기로 한다. 즉, 유전자 i 로부터 유전자 j 까지의 유전자거리 $D=(d_{ij})$ 는 다음과 같이 지정된다.

$$d_{ij} = |\{l \in V : c_{il} < c_{ij}\}| \quad (2)$$

여기서 $V=\{1, 2, \dots, n\}$ 는 노드들의 집합을 나타내고 c_{ij} 는 비용행렬 C 의 (i, j) 원소를 의미한다. c_{ij} 의 값이 작을수록 d_{ij} 의 값도 작아짐을 알 수 있다. 즉, 두 유전자간의 유전자 거리는 거기에 대응되는 노드들 간의 비용이 상대적으로 작을수록 작은 값을 가진다. 따라서 염색체공간 상에서 상대적으로 “가까운” 곳에 위치하게 된다.

3.4 이질적 교배

보로노이양자화교차는 다른 교차들에 비해서 빠른 수렴속

도를 가지는 것으로 알려져 있다. 따라서 해집단이 성숙되지 못하고 수렴하는 경우가 발생할 수 있다. 이것을 피하기 위해서 Seo와 Moon[7]은 특수한 형태의 교배방법을 사용하였다. 이질적 교배라고 부르는 이 교배방법은 첫 번째 부모해를 선택한 다음 두 번째 부모해를 선택할 때 첫 번째 부모해와 유전적으로 상이한 해가 선택될 확률이 서로 유사한 해가 선택될 확률보다 높게 만들어준다. 즉, 첫 번째 부모해 p_1 이 선택되었을 때 두 번째 부모해의 후보를 m 개 선택하여 이들 중 p_1 과 유전적으로 가장 다른 해를 두 번째 부모해로 선택한다. 이 때, 해들 간의 거리측도로는 해밍거리[2]를 사용한다.

4. 실험결과

본 논문에서 사용된 유전알고리즘은 새로운 해가 만들어질 때마다 해집단에 대치되는 안정상태 혼합형 유전알고리즘이다. 실험에 사용된 연산자들과 파라미터들을 요약하면 다음과 같다.

- 해집단의 초기화 - 주어진 해집단의 크기만큼의 해들을 임의로 생성한 다음 지역탐색휴리스틱을 적용한다. 해집단은 모두 가능해들로 구성된다.
- 해집단의 크기 - $|P|=50$.
- 선택연산자 - 룰렛휠선택이 사용된다. 즉, 해집단 내의 최선해의 선택확률은 최악해의 선택확률보다 5배 큰 값이 된다.
- 이질적 교배에서 후보의 개수 - $m=3$.
- 보로노이양자화교차에서 중심유전자의 개수 - $k=6$.
- 변이연산자 - 선행관계조건을 위반하지 않는 경로보전 3-교환 중 하나를 임의로 골라서 적용하는 것을 5회 반복한다. 이러한 변이는 각 자식해에 대해서 0.1의

2) 두 해 사이의 값이 서로 다른 유전자의 개수.

확률로 적용된다.

- 지역탐색휴리스틱 - 경로보전 3-최적화[3]가 사용된다. 여기서는 주어진 해에 대해서 선행관계조건을 위반하지 않으면서 최대 이득을 갖는 경로보전 3-교환을 찾아서 적용하는 과정이, 더 이상의 개선이 없을 때까지 반복된다.

표 1. 실험결과 요약.

Table 1. Summary of the experimental results.

	DGA	EGA	MGA	VGA
BBI#	14	13	14	18
BAI#	6	7	1	17

- 대치연산자 - Bui와 Moon[12]이 사용한 것과 유사한 대치연산자가 사용된다. 즉, 각각의 자식해는 1) 만일 자식해가 두 부모해 중 더 유사한 부모해보다 높은 적합도를 가지면 그 부모해와 대치되고, 2) 그렇지 않고 두 부모해 중 덜 유사한 부모해보다 높은 적합도를 가지면 그 부모해와 대치되고, 3) 그렇지 않으면 해집단 내의 최악해와 대치된다.
- 종료 조건 - 실험에 사용된 서열순서화문제 인스턴스들 중 prob.100을 제외한 나머지 인스턴스들에 대해서는, 10000세대 동안 해집단 내의 최선해가 개선되지 않으면 종료하도록 하였다. prob.100은 다른 인스턴스들보다 어려운 문제이기 때문에 해집단 내의 최선해가 100000세대 동안 개선되지 않으면 종료하도록 하였다.

모든 프로그램은 Pentium IV 1GHz 시스템에서 운영되는 리눅스 환경에서 C언어로 구현되었다. 실험은 TSPLIB³⁾와 ZIB-MP-Testdata⁴⁾로부터 얻은 18개의 서열순서화문제 인스턴스들을 대상으로 진행되었다. 이 인스턴스들은 노드의 수가 70개 이상인 인스턴스들을 모두 모은 것이다.

본 실험에서는 서열순서화문제를 위해서 제안된 혼합형 유전알고리즘의 성능을 평가하기 위해서 세 가지 다른 교차연산자를 사용하는 유전알고리즘들과 성능을 비교하였다. 비교의 공정성을 위해서 교차연산자를 제외한 모든 연산자들과 파라미터들을 통일시켰다. 비교에 사용된 교차연산자들은 거리보전교차(Distance-Preserving Crossover; DPX)[14], 간선재결합(Edge Recombination; ER)[15], 최대부분순서/임의삽입(maximum partial order/ arbitrary insertion; MPO/AI)[4] 등이다. 최대부분순서/임의삽입은 두 부모해의 최대 공통 부분수열을 자식해로 복사한 다음 아직 경로에 포함되지 않은 노드들을 임의의 순서로 골라서 삽입 가능한 위치들 중 최소의 비용을 가지는 곳에 삽입한다[4]. 실제 구현은 Chen과 Smith의 구현⁵⁾을 그대로 사용하였다. 거리보전교차와 간선재결합은 원래 순회판매원문제를 위해서 개발된 교차연산자이기 때문에 서열순서화문제의 선행관계조건을 만족시키도록 수정하여 사용하였다. 거리보전교차는 두 부모해에 대해서 같은 해밍거리를 갖는 자식해를 만들어낸다. 즉, 자식해

o 와 부모해 p_1 , 자식해 o 와 부모해 p_2 및 부모해 p_1 과 부모해 p_2 사이의 해밍거리들이 모두 같도록 하는 자식해 하나를 생성한다[14]. 간선재결합은 먼저 두 부모해에 대한 간선지도를 작성하는데, 이것은 각 노드에 대해서 두 부모해의 해밀턴경로 상에서 그 노드와 연결된 노드들을 리스트로 저장한 것을 말한다. 간선재결합은 임의로 선택된 노드에서 출발하여 간선지도를 참조하여 부모해의 해밀턴경로 상에서 현재의 노드와 연결된 적이 있는 노드들 중 하나를 골라 연결하는 것을 반복한다. 이 때, 연결을 위한 후보 노드는 최대 4개가 될 수 있는데, 이들 중 자신으로부터 연결 가능한 후보 노드를 가장 적게 가지고 있는 노드를 고른다[15].

표 2. VGA에 의해서 갱신된 Best-Known 결과들.

Table 2. New best-known records by VGA.

인스턴스	기존의 Best-Known	갱신된 Best-Known
prob.100	1190	1145
rbg323a	3141	3140
rbg341a	2570	2568

표 1은 실험결과를 요약한 것인데, 각 유전알고리즘에 대해서 그 유전알고리즘의 최선의 결과가 다른 알고리즘에 비해서 뒤지지 않는 인스턴스의 개수(BBI#)와 그 유전알고리즘의 평균 결과가 다른 유전알고리즘에 비해서 뒤지지 않는 인스턴스의 개수(BAI#)를 각각 표시하였다. DGA, EGA, MGA, VGA는 거리보전교차, 간선재결합, 최대부분순서/임의삽입 및 보로노이양자화교차를 각각 사용하는 유전알고리즘을 의미한다. 각 인스턴스의 난이도와 수행시간 등을 고려하여 ESC78, ft70*, kro124p*, rbg109a 등은 100회, prob.100, rbg150a, rbg174a, rbg253a, rbg323a 등은 20회, rbg341a, rbg358a, rbg378a 등은 10회 실행하였다. 표에 나타난 결과를 보면 VGA가 비교에 참여한 다른 유전알고리즘에 비해서 더 안정적이고 성능이 우수함을 알 수 있다. 표 2는 VGA에 의해서 갱신된 Best-Known 결과들을 정리한 것이다. 그림 6은 실험에 사용된 18개의 인스턴스들 중 대표적인 4개에 대한 평균 결과를 그래프로 나타낸 것이다.

5. 결 론

본 논문에서는 서열순서화문제를 풀기 위한 새로운 혼합형 유전알고리즘을 제안하였다. 제안된 유전알고리즘은 보로노이양자화교차를 주된 연산자로 사용한다. 이 교차는 원래 순회판매원문제를 위해서 제안된 교차연산자로서, 이를 서열순서화문제에 적용하기 위해서는 상당한 변형이 요구된다. 본 논문에서는 서열순서화문제에 맞도록 변형된 보로노이양자화교차(3.2.2절)와, 이 때 필요한 가능해생성알고리즘(3.2.3절), 선행관계사이클분해알고리즘(3.2.4절), 유전자거리지정방법(3.3절) 등을 제안하였다. 제안된 혼합형 유전알고리즘에서는 경로보전 3-최적화를 지역탐색휴리스틱으로 사용한다.

제안된 유전알고리즘의 성능을 평가하기 위해서 세 가지 다른 교차연산자인 거리보전교차, 간선재결합 및 최대부분순서/임의삽입을 각각 사용하는 유전알고리즘들과 비교하였다. 비교실험은 TSPLIB와 ZIB-MP-Testdata로부터 얻은 크기가 가장 큰 18개의 인스턴스들에 대해서 진행되었다. 실험결과, 제안된 유전알고리즘이 비교된 다른 유전알고리즘에 비

3) TSPLIB <http://www.iwr.uni-heidelberg.de/groups/comopt/software/TSPLIB95/>

4) ZIB-MP-Testdata <http://elib.zib.de/pub/Packages/mp-testdata/tsp/sop/index.html>

5) MPO/AI http://www.cs.cmu.edu/afs/cs.cmu.edu/user/chens/WWW/MPOAI_SOP.tar.gz

해서 더 안정적이고 성능이 우수한 것으로 나타났다. 향후 연구방향은 서열순서화문제를 풀기 위한 더 강력한 연산자들을 개발하는 것이다.

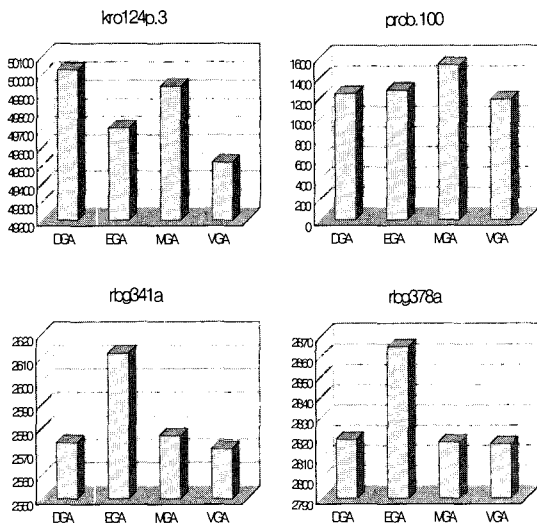


그림 6. 성능비교 그래프.

Fig. 6. Performance Comparison Graphs.

참고 문헌

[1] L. F. Escudero, "An exact algorithm for the sequential ordering problem," European Journal of Operational Research, Vol. 37, pp. 232-253, 1988.
 [2] M. R. Garey and D. S. Johnson, Computers and Intractability: A Guide to the Theory of NP-Completeness, W. H. Freeman and Company, 1979.
 [3] L. M. Gambardella and M. Dorigo, "An ant colony system hybridized with a new local search for the sequential ordering problem," INFORMS Journal on Computing, Vol. 12, No. 3, pp. 237-255, 2000.
 [4] S. Chen and S. Smith, Commonality and genetic algorithms, Technical Report CMU-RI-TR-96-27, The Robotics Institute, Carnegie Mellon University, 1996.
 [5] 이건명, 이혜리, "서열 순서화 문제와 job shop 문제에 대한 선행관계유지 유전연산자의 비교," 퍼지 및 지능시스템 학회 논문지, 제14권, 5호, pp. 563-570, 2004.
 [6] 이혜리, 이건명, "Sequential ordering problem과 job shop scheduling problem에 적용 가능한 선행관계유지 유전 연산자의 비교," Journal of the Research Institute for Computer and Information Communication, Vol. 7, No. 2, 1999.
 [7] D. I. Seo and B. R. Moon, "Voronoi quantized crossover for traveling salesman problem," In Genetic and Evolutionary Computation Conference, pp. 544-552, 2002.
 [8] J. Holland, Adaptation in Natural and Artificial Systems, The University of Michigan Press, 1975.
 [9] D. E. Goldberg, Genetic Algorithms in Search, Opt

imization, and Machine Learning, Addison-Wesley, 1989.
 [10] 문병로, 유전알고리즘, 두양사, 2003.
 [11] D. I. Seo and B. R. Moon, "A survey on chromosomal structures and operators for exploiting topological linkages of genes," In Genetic and Evolutionary Computation Conference, pp. 1357-1368, 2003.
 [12] T. N. Bui and B. R. Moon, "A new genetic approach for the traveling salesman problem," In IEEE Conference on Evolutionary Computation, pp. 7-12, 1994.
 [13] A. Gersho and R. M. Gray, Vector Quantization and Signal Compression, Kluwer Academic Publishers, 1992.
 [14] B. Freisleben and P. Merz, "New genetic local search operators for the traveling salesman problem," In Parallel Problem Solving from Nature, pp. 890-900, 1996.
 [15] D. Whitley, T. Starkweather, and D. Fuquay, "Scheduling problem and traveling salesman: the genetic edge recombination operator," In International Conference on Genetic Algorithms, pp.133-139, 1989.

저자 소개



서동일(Dong-II Seo)

1996년 : 한국과학기술원 전산학(학사)
 1998년 : 한국과학기술원 전산학(석사)
 1998년~현재 : 서울대학교 컴퓨터공학부 박사과정

관심분야 : 알고리즘설계, 유전알고리즘, 정보이론, 확률과정
 E-mail : diseo@soar.snu.ac.kr



문병로 (Byung-Ro Moon)

1985년 : 서울대학교 계산통계학(학사)
 1987년 : 한국과학기술원 전산학(석사)
 1994년 : Pennsylvania주립대 전산학(박사)
 1997년~현재 : 서울대학교 컴퓨터공학부 부교수

관심분야 : 최적화, 알고리즘설계, 유전알고리즘, 진화연산, 복잡계
 E-mail : moon@soar.snu.ac.kr