경쟁부분 이론분야 논문

직각거리 스타이너 나무 문제의 하이브리드 진화 해법에서 효율적인 적합도 추정에 관한 연구

양병학

경원대학교 산업정보시스템공학과

An Estimation of Fitness Evaluation in Evolutionary Algorithm for the Rectilinear Steiner Tree Problem

Byounghak Yang

Industrial Engineering, Kyungwon University

Abstract. The rectilinear Steiner tree problem is to find a minimum-length rectilinear interconnection of a set of terminals in the plane. It is well known that the solution to this problem will be the minimal spanning tree (MST) on some set Steiner points. A hybrid evolutionary algorithm is introduced based upon the Prim algorithm. The Prim algorithm for the fitness evaluation requires heavy calculation time. The fitness value of parents is inherited to their child and the fitness value of child is estimated by the inherited structure of tree. We introduce four alternative evolutionary algorithms. Experiment result shows that the calculation time is reduced to 25% without loosing the solution quality by using the fitness estimation.

Keyword: Rectilinear Steiner tree problem, Evolutionary Algorithm, Fitness estimation

1 서론

<그림 1> 직각거리 최소비용 나무와 직각거리 스타이너 나무

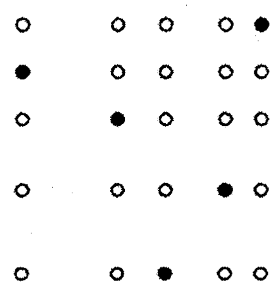
반도체를 설계하거나 물류 설비를 설계할 때 직각거리로 모든 점을 연결해야하는 문제들이 있다. 단순히 주어진 모든 기본점만을 연결한다면 이는 최소비용나무(MST: Minimum Spanning Tree)문제이다. 그런데 주어진 기본점 외에 비용을 절감하기위해 중간 연결점 또는 스타이너점을 추가하면 더 싼 비용의 나무를 구성할 수 있다. 이와 같은 문제를 평면상의 스타이너 문제(STP: Steiner Tree Problem)라고 하며, 평면상에 n개의 기본점에 의한 집합 V가 주어졌을 때, V에 속한 모든 점을 연결하는 최소비용나무를 구하되 임의의 추가점을 허용하는 문제이다. 그림1에서 4개의 기본점을 최소 비용으로 연결한다면 그림1.a와

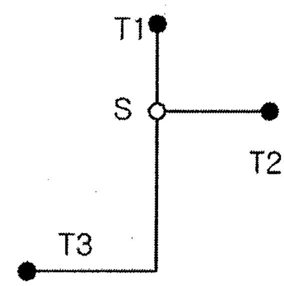
같은 형태로 연결하는 것이 최소비용이다. 그러나 만약 그림1.b와 같이 적절한 추가 연결점을 사용하면 전체 비용은 절감될 수 있다. 이때 임의의 추가점을 스타이너점이라 하고 그 집합을 S라 한다. 만약 스타이너점의 추가를 허용하지 않는다면 이 문제는 단순한 최소비용나무 문제가 된다. 또한 S가 확정된 상태에서의 STP의 해는 V와 S의 합집합(V∪S)에 대한 MST로 구할 수 있음은 잘 알려져 있다. 스타이너점의 예이다.

MST(V)를 기본점집합 V를 연결하는 최소비용나무의 값이라 하자. 그러면 스타이너나무의 절약값은 다음과 같이 정의된다.

R(V) = {MST(V∪S) - MST(V) } / MST(V).

이 절약값은 스타이너 나무 해법의 효율성을 측정하는 단위로 사용된다. Warme, Winter and Zachariasen [25] 은 RSTP를


(a)



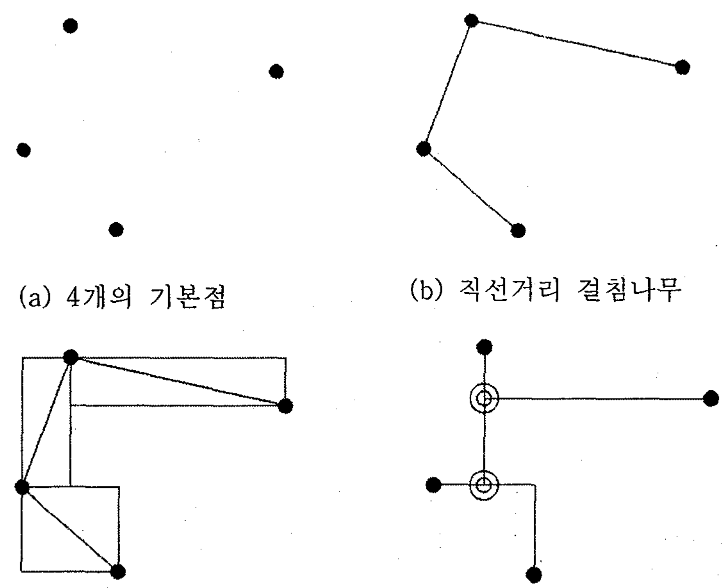
(b)

<그림 2> Hanan격자와 3개의 기본점에 대한 최적스타이너점

이너 문제의 종류로는 크게 평면상의 스타이너 문제와 그래프상의 스타이너 문제 (GSTP : Graphical Steiner Tree Problem)로 나뉜다. GSTP에서는 미리 후보 스타이너점들이 주어지는데 이 경우 후보 스타이너점들 중 최적 스타이너점들의 조합을 찾는 문제가 된다. 반면에 평면상의 스타이너 문제는 스타이너점들이 주어지지 않아서 스타이너점의 위치와 스타이너 나무를 동시에 구성해야 한다. 이때 점과 점사이의 거리가 직선거리인 스타이너 문제 (ESTP : Euclidean Steiner Tree Problem)와 직각거리인 직각거리 스타이너 문제(RSTP: Rectilinear Steiner Tree Problem)로 나뉜다. 우리는 그중 RSTP문제를 다루려고 한다.

RSTP는 이미 NP-complete으로 알려져 있다[8]. 최적해를 지수함수 시간대로 풀 수 있는 해법은 존재 하지 않는 것으로 알려져 있다[7]. Hanan은 모든 직각거리 스타이너 나무에서 최적 스타이너점은 기본점들을 지나는 수직 수평선으로 이루어진 격자 공간의 교점상에 존재함을 보였다[10]. Hanan의 정리는 Hanan격자(grid)라는 이산적인 공간에 직각거리 스타이너점이 존재한다는 것을 의미하며 이런 이산적인 성질이 ESTP의 연속적인 성질과 가장 차이가 나는 점이다. Hanan 격자의 예는 [그림2.a]와 같다. 기본점이 3개인 경우에는 최적스타이너점을 쉽게 구할 수 있다. (x_i,y_i)를 주어진 기본점 Ti의 공간상 위치라 하자. France [6]는 스타이너점 S는 (x_m,y_m)에 존재하며, x_m 과 y_m 은 각각 { x_i }와 { y_i}의 메디안이다. [그림2.b]는 3개의 기본점에 대한 최적 위한 최적해법을 제시 했는데 그들의 연구결과에 의하면 표준 문제의 평균 절약값이 11.4% 정도였다. 표준 문제는 Beasley[3]에 의해서 제시 되었으며, 그는 간단한 휴리스틱 해법도 소개하였다[4]. 대부분의 휴리스틱은 최소비용나무를 이용하고 있다. Ho, Vijayan, and Wong [12] 은 최소비용나무를 직각사각형 영역들로 나누고 L-형과 Z-형 구조를 이용하여 근사해를 구하였다. Lee, Bose and Hwang [21] 은 최소비용나무 해법인 Prim해법을 변형한 휴리스틱 해법을 제시하였다. Hwang[13] 역시 최소비용나무를 이용하여 계산 시간을 단축한 휴리스틱을 제시하였다. 휴리스틱 해법 중 가장 우수한 것들은 Kahng과 Robins[20]이 제시한 the Batched Iterated 1-Steiner(BI1S)방법이다. 그들은 주어진 기본점들로 이루어진 최소비용나무로부터 출발하여 비용을 최소화하는 스타이너점을 하나씩 탐색하여 나무에 추가시켰다. 다른 방법으로는 Boar와 Owens[5]가 제시하였다. 두 방법 모두 평균 절약값은 10.7%정도였다.

스타이너 나무 문제를 위한 진화 해법들을 문제별로 찾아보면 다음과 같다. 먼저 ESTP에 대한 연구로는 Hesser[11], Jones과 Harris[14], Barreiros[2]와 Yang[] 등이 있다. Hesser[11]는 스타이너점들의 2차원상의 위치인 (x, y)로 이루어진 집합으로 개체를 구성하였다. 최적 스타이너점의 수를 알 수 없음으로 미리 정한 수만큼의 스타이너점등을 계속 유지해 나가는 전략이었다. Jones과 Harris[14]는



(a) 4개의 기본점 (b) 직선거리 걸침나무

(c) 걸침나무로부터 구한 가능한 직각 에지 (d) 걸침나무로부터 파생된 직각거리 스타이너 나무

<그림 3> 직선거리 걸침나무(spanning tree)를 이용한 직각거리 스타이너 나무의 구성 방법

(x,y)집합과 각 스타이너점이 유효한지 무효인지를 정해주는 자료구조를 객체로 이용하였다. 그러나 해의 품질은 우수하지 못했다. 그들은 그 이유를 유전 해법의 우수한 스타이너점을 찾아내는 방법이 미약했다고 지적했다. Barreiros [2]는 소규모 문제를 위한 유전해법으로 '척추(backbone)'라는 개념을 도입했다. 척추란 모든 스타이너점이 이웃 스타이너점과 직접 연결되어 마치 척추와 같은 형태의 구조를 가지고 있다. 기본점은 척추의 스타이너점 중 가장 근접한 스타이너점에 연결시켰다. 원 문제를 풀기위해 먼저 기본점들을 지역적으로 분할하여 소규모 문제로 만들고 소규모 문제는 이 척추를 이용한 유전해법을 수행하였다. 원래 문제의 해는 소규모 문제의 해들을 결합하여 구성하였다.

다음은 RSTP에서의 진화해법을 알아보자. <그림3>을 보면 먼저 4개의 기본점이 있다. 이 4개를 연결하는 걸침 나무를 <그림3.b>라 하자. 그러면 걸침나무의 한 에지에 대하여 직각으로 연결하는 방법은 두 가지가 가능하다. <그림3.c>에 가능한 조합을 표시하였다. 각 에지 마다 가능한 직각 에지를 선택하면 가능한 하나의 직각거리 스타이너 남무를 구성할 수 있다. 이러한 성질을 응용한 유전해법들이 개발되었는데 그들은 먼저 <그림3.b>와 같은 걸침나무를 유전해법에서 개체로 사용하는 방법에 대하여 연구하였다. 초기의 연구로는 Prüfer수를 이용하여 걸침나무를 형성하는 방법이 사용되었다[16]. Prüfer수가 진화 해법에서 우수하지 못한 성질이 있어서 이를 극복하기 위한 특별한 교차와 돌연변이 연산을 도입한 연구가 있었지만 해의 품질은 좋지 않았고 문제의 크기도 기본점 32개짜리를 해결하였다[9]. 좀 더 효율적인 유전해법은 Palmer와 Kershenbaum[22]가 제시한 유전 해법에서의 나무 표현 방법을 Julstrom[17]이 RSTP로 확장한 가중치 방법과 에지 나열법[18]에 기반을 둔 진화 해법들을 제시하였다. 그는 세 가지 방법을 비교 분석하였는데 에지 나열법이 상대적으로 해의 품질이 우수하였다. 이때 해결한 문제의 크기는 200개 정도였으나 해의 품질은 상당히 저조했다. Julstrom[19]은 직선거리 최소비용나무를 구하고, 이 직선거리 최소비용나무에 대하여 에지에서 스타이너점을 선택하는 유전 해법을 도입하여 문제의 크기를 500개까지로 확장했었다. 이 연구 결과는 해의 품질도 그전의 연구보다는 상당히 우수하였다. Wakabayashi[24]는 기본점에 대한 라벨 집합과 스타이너점을 구성하기 위한 '+' 기호로 이루어진 자료구조를 개체로 이용한 다목적함수 RSTP에 대한 유전 해법을 제시하였다.

Yang[21,22]은 기존의 연구들과는 달리 스타이너점 자체를 개체로 표현하고 스타이너점의 공간상의 위치를 개체로 하는 진화 해법을 제시하였다. 그의 연구에 의하면 평균 절약값이 11.0%로 기존의 휴리스틱보다 좋았으며

거의 최적해에 근접하였다. 그러나 개체를 평가하기 위해서 Prim해법에 의한 최소비용나무를 풀어야 하기 때문에 개체의 평가 시간이 많이 걸린다는 약점이 있었다.

진화해법에서 개체의 적합도를 평가하기 위해 요구되는 연산의 어려움은 많은 연구에서 지적되고 있다. 계산 시간을 절약하기 위한 방법으로 적합도 추정법을 사용할 수 있다. 최적화 문제에서 최적값 추정은 이미 알려진 개념이다. 전통적으로 두 가지 추정법이 존재하는데, 첫 번째는 문제의 추정법이다. 예를 들어 복잡한 정수계획법문제를 단순한 선형계획법으로 추정하여 목적함수값의 상하한을 구하는 것이 그 예이다. 두 번째 방법은 목적함수값 추정법이다. 복잡한 목적함수대신 단순한 목적함수를 사용하는 것이다. 예를 들어 개체를 평가하기 위해 복잡한 시뮬레이션을 수행해야 하는 경우, 간단한 대기모형식으로 대신하는 경우이다. 세 번째는 진화 해법에서 사용될 수 있는 개념으로 진화적 추정이다. 이것은 부모의 개체값을 자식이 물려받고 이를 이용하여 자시의 개체값을 추정하는 것이다[15]. 본 연구에서는 진화적 추정 개념을 도입하여 적합도 검정의 연산 시간을 절감하려고 한다.

2 진화 해법

진화해법은 생물의 진화과정(자연선별과 돌연변이)을 모방한 확률적 탐색기법이다. 진화해법의 가장 큰 특징은 고전적 해법들이 하나의 해를 유지하며 이를 개선해 가는 과정이지만 진화해법은 복수개의 잠재해로 이루어진 해집단을 운용하는 것이다. 진화해법은 문제의 잠재해를 표현한 개체들로 이루어진 모집단을 가지고 시작한다. 모집단은 매 세대마다 일정수의 개체를 유지하고 매 세대마다 각 개체의 적응도를 평가하여 다음 세대에 생존할 개체들을 확률적으로 선별한다. 선별된 개체들 중 일부의 개체들이 서로 교배하여 새로운 개체를 생성하고, 일부개체는 돌연변이를 통해 변경한다. 즉, 선별을 통해 우수한 개체를 다음 세대로 상속시키고, 교배와 돌연변이를 통해 새로운 해를 탐색하는 과정이다. 이러한 진화과정을 종료조건이 만족될 때까지 반복하여 해법 진행 중 발생한 최우수해를 찾아내는 것이 진화해법이다. 진화해법은 그 개념과 이론이 단순하고 해의 탐색성능이 우수하여 많은 최적화 문제에 다양하게 적용하고 있다[1]. 본 연구에서 사용할 기본적인 RSTP에 대한 진화 해법은 이미 제시된 연구 결과에 자세히 기술되어 있어서 [21,22,23] 본 논문에서는 간단하게 기술하겠다.

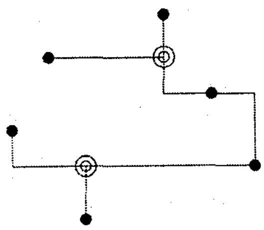
2.1 RSTP를 위한 일반 진화해법

개체: RSTP에서 스타이너점들은 Hanan 격자에 존재한다. 우리는 스타이너점의 수와 스타이너점의 Hanan 위치를 개체로 정의하였다. 즉, 각 개체는 {m, (v_1, h_1), (v_2, h_2), ... , (v_m, h_m) }로 표현되며, m은 스타이너점의 수, (v_i, h_i)는 i번째 스타이너점 S_i 의 Hanan격자상의 좌표이다.

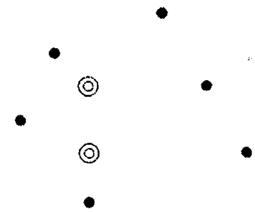
교차 연산: 선택된 두 부모의 각 스타이너점들을 랜덤하게 선택하여 이산 교차를 수행하였다.
돌연변이 연산 : 랜덤하게 선택된 스타이너점의 위치(v_i, h_i)를 랜덤함수에 의해 (v_i+v, h_i+h)로 변경하였다. 여기서 v와 h 는 랜덤 함수값이다.
선별: 선별은 토너먼트 크기가 2인 토너먼트 선별을 사용하였다.
초기해: 기본점만으로 연결된 최소비용나무를 구성하고 이 나무 상에서 직접 연결된 3개의 기본점 집합들을 모두 구했다. 그리고 3개의 기본점에 대한 최적 스타이너점을 구했다. 이 3개의 기본점을 위한 스타이너점은 원래 문제의 최적스타이너점은 아니지만 최적해에서 살아남을 확률이 높을 것으로 기대했다. 그래서 3개의 기본점에 대한 스타이너점을 모두 구하여 그 중에서 랜덤하게 선택한 스타이너점들로 초기 개체들을 구성하였다.
개체 적합성 평가: 주어진 스타이너점과 기본점을 연결하는 최소비용나무를 구한다.

2.2 혼합 연산자(Hybrid operator)

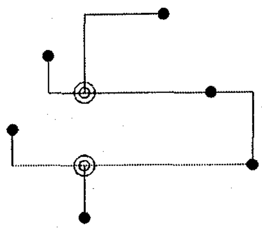
진화 해법에 해의 탐색을 강화하기 위해 세 가지 지역 탐색을 도입했다. 첫 번째는 임의의 점을 스타이너점으로 추가하는 삽입전략이다. 일반적인 진화해법에서는 스타이너점들의 위치만을 탐색하게 되는데 이런 경우 더 좋은 스타이너점이 추가로 존재하는 경우에 탐색이 쉽지 않다. 그래서 우리는 선택된 개체에 대하여 Hanan 격자 상에서 랜덤하게 발생시킨 스타이너점을 추가하는 삽입 전략을 사용하였다. 두 번째는 명백하게 불리한 스타이너점을 제거하는 제거전략이다. 스타이너점의 차수, 즉 스타이너점과 연결된 다른 점의 수가 2개 이하라면 이는 명백히 불리한 스타이너점이다. 우리는 랜덤하게 선정된 개체에 대하여 이러한 불리한 스타이너점을 탐색하여 강제로 제거하는 제거 전략을 사용하였다. 세 번째는 스타이너점 개선 전략이다. 스타이너점과 연결된 나머지점이 3개인 경우를 고려해 보자. 이 경우에 3개의 점을 연결하는 최적 스타이너점은 쉽게 구할 수 있고 스타이너점을 그 최적 점으로 이동시키면 나무의 비용을 절약할 수 있다. 랜덤하게 선택된 개체에 대하여 스타이너점의 개선 전략을 사용했다.



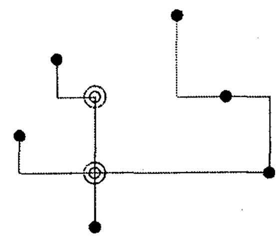
(a) 부모 개체와 최소비용나무



(b) 자식 개체



(c) 부모의 나무 구조를 상속받은 자식 개체



(d) 자식개체의 최소비용나무

<그림 4> 개체 적합도 평가를 위한 최소비용나무와 추정

2.3 적합도 추정

개체의 평가를 위해서 우리는 주어진 스타이너점들과 기본점들을 연결하는 최소비용나무를 구성해야 한다. 이는 모든 개체에 대하여 최소비용나무를 구해야하고 이로 인하여 RSTP의 진화해법이 우수한 품질의 해를 제공함에도 계산 시간에서는 불리한 것으로 보고되는 이유이다. 이를 해결하기 위해 우리는 적합도에서 추정 방법을 도입하기로 했다. [그림4.a]에서 하나의 부모 개체와 그 개체에 대한 최소비용나무가 구해져 있다. 진화해법의 여러 연사자에 의해서 그 자손 개체의 스타이너점이 이동하였다고 하자. 그 결과가 [그림4.b]이다. 이 개체에 대한 정확한 평가를 위해서는 최소비용나무를 구해야하는데 그 결과가 [그림4.d]이다. 최소비용나무를 구하는 계산 복잡도는 O(*nlogn*)으로 알려져 있다. [그림4.a]와 [그림4.d]의 나무구조는 서로 다름을 알 수 있다. 그런데 만약 [그림4.a]의 나무 구조를 그대로 이용하여 자식 개체를 위한 나무를 구성하면 [그림4.c]가 된다. [그림4.c]의 나무는 [그림4.a]의 나무와 각 점의 연결 구조가 동일하다. 따라서 이동한 스타이너점과 직접 연결된 가지들의 비용만 수정하면 쉽게 나무의비용을 구할 수 있다. 이때 필요한 계산의 복잡도는 변경된 스타이너점과 그에 연결된 에지의 수에 비례한다. 따라서 연산 시간에서 상당한 절감이 예상된다. 반면 정확한 자식 개체의 평가값이 아닌 추정값을 사용하게 된다. 우리는 [그림4.c]와 같은 부모 개체의 나무 구조를 그대로 상속받은 자식 개체의 나무 비용으로 자식개체를 평가하였다. 이 방법은 진화 해법이 진행됨에 따라 많은 개체가 추정값을 사용하게 되고, 추정을 반복적으로 많이 한 개체는 오차가 점점 커지게 될 수 있다. 이를 해소하기위해 다음 두 가지 방법을 사용하였다. 먼저 모집단 중 최우수개체에 대하여는 적합도 추정값이 아닌 최소비용나무를 구해서 정확한 개체 평가값을 사용하도록 하였다. 두 번째로 진화해법을 수행하던 중간에 해의 개선이 이루어지지 않아서 해법진행을 종료하기 직전 상황이 되면 모든 개체에 대하여 최소비용나무를 구해서 추정값이 아닌 정확한 개체값으로 개체를 평가하였다.

2.4 진화해법의 대안

우리는 이제 네 가지 진화 해법의 대안을 다음과 같이 제시하였다.

EA(Evolutionary Algorithm) 진화 해법만을 사용하고, 지역탐색이나 적합도 추정을 사용하지 않음

EA_H(Hybrid EA) 지역탐색만을 사용하는 진화해법

EA_E(Estimation EA) 적합도 추정만을 사용하는 진화해법

EA_HE(Hybrid Estimation EA) 적합도 추정과 지역 탐색을 모두 사용하는 진화 해법

[그림5]에는 혼합진화해법의 절차를 예시하였다. 이 절차에서 적합도 검정 방법은 적합도 추정을 사용하는 경우와

최소비용나무를 이용하는가에 따라 다를 수 있다.

```
HES (Hybrid Evolutionary Strategy)

begin
  t = 0
  Initialize population P(t)
  Fitness evaluation P(t)
while (do not satisfy termination criteria)
do
  begin
    t = t +1
    Selection P(t) from P(t-1)
    Crossover on P(t)
    Mutation on P(t)
    Local search on P(t)
    (Estimated) Fitness evaluation P(t)
    end
end
```

<그림 5> 혼합진화전략의 절차

3 실험 결과

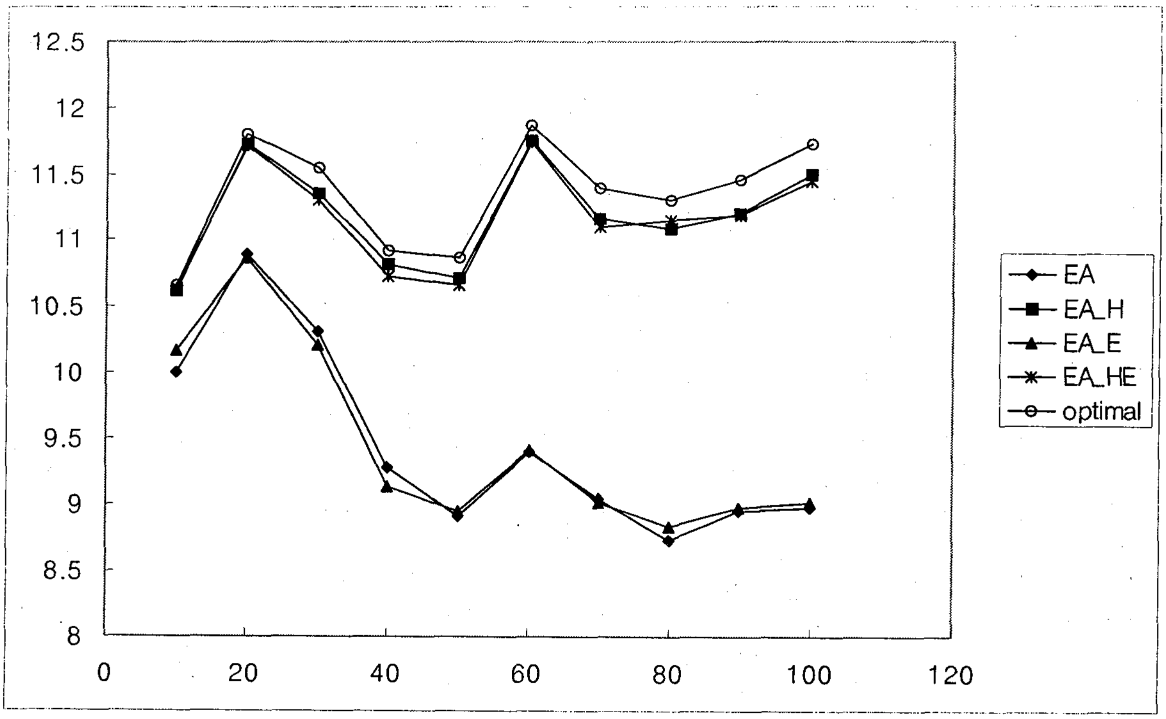
실험은 Pentium IV급 컴퓨터에서 수행되었다. 실험문제는 OR-Library [3]에서 제시된 표준 문제로 기본점의 수가 10개, 20개, ……, 100개, 250개, 500개 및 1000개이고 각각 15개씩으로 구성되어 있다. 먼저 기본점의 수가 10개에서 100개까지의 문제들에 대한 진화해법간의 비교 실험을 실시하였다. <그림6>에 네가지 진하해법으로 구한 해와 Warme[26]이 제시한 최적해와의 비교 결과이다. 진화해법간의 비교에서 EA_H와 EA_HE의 결과가 EA나 EA_E보다 우수하였다. 이는 지역탐색 전략이 해의 품질을 개선한다는 것을 의미한다. 특히 최적해와의 비교에서 EA_H와 EA_HE의 결과가 최적해와 유사하게 나옴을 알 수 있다. <그림7>은 진화해법간의 연산시간의 비교분석이 나와 있다. 연산시간에서는 EA_E나 EA_HE가 훨씬 우수함을 알 수 있다. 적합도검정에서 추정법을 사용함으로써 연산시간의 절감효과가 있음을 확인할 수 있다. 종합적으로 보면 EA는 해의 품질과 연산 시간 면에서 모두 열등하였다. EA_E의 경우는 해의 품질이 열등하였다. EA_H의 경우에는 해의 품질은 우수하였으나 연산 기간이 열등하였고 EA_HE가 해의 품질과 연산 시간에서 우수한 경과가 나왔다.

기존의 유전 해법과 본 연구에서 제시한 진화 해법간의 비교를 위해서 기존의 유전 해법 연구 중 가중 우수한 것으로 알려진 Julstrom의 2002년 연구 결과 [22]와 비교했다. Besealy가 제시한 표준 문제 중 문제의 크기 100,250,500개 인 경우에 한 문제당 50번의 유전 해법을 수행한 후 그 중에서 가장 우수한 해들을 선택하였다. 그 결과가 <표1>에 제시되었다. 우리는 HE_H와 HE_HE에 대하여 진화 해법을 한 번만 수행한 후 구해진 해들을 <표1>에 제시하였다. 최적해는 Warme[26]이 제시한 결과를 이용하였다. 각 해법에 의해 구해진 나무의 비용을 제시하고, 최적해와의 비교를 위해 최적해대비 증가한 나무비용을 증가율로 표시하였다. <표1>을 모든 비교 문제에 있어서 HE_H와 HE_HE가 기존의 유전 해법의해보다 우수함을 알 수 있다. Julstrom의 해가 최적해와 평균 2.79% 정도의 차이를 나타내고 있으나 진화해법들은 평균 0.6% 정도의 차이를 나타내고 있어서 최적해에 근접했음을 알 수 있다.

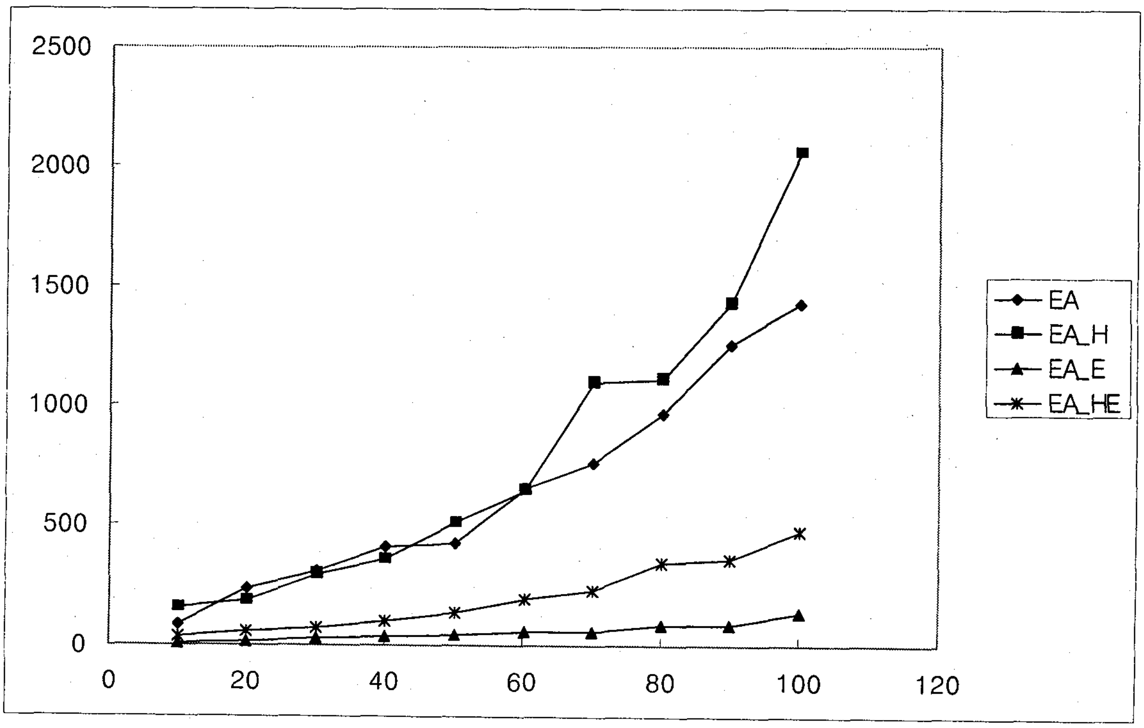
<표2>에서는 기존에 나와 있던 휴리스틱들과 비교하였다. 역시 OR-library에서 제시된 표준 문제들을 각 연구자들이 자신의 해법으로 해결하였고, 각각의 논문에서 절약값의 결과를 추출하였다. HE_E와 HE_HE가 기존의 다양한 휴리스틱과 비교하여도 절약값에 있어서 더 우수함을 알 수 있었다. <표2>의 최적해와 비교해보면 진화해법이 최적해에 근접해 있음을 알 수 있다.

4 결론

본 논문에서 직각거리 스타이너 나무 문제를 위한 혼합 진화 해법을 제시하였다. 진화 해법을 도입함에 있어서 개체는 Hanan 격자상의 스타이너점들의 위치 집합을 사용하였다. 진화 연산자로는 이산 교차와, 돌연변이 그리고 토너먼트 선별이 사용되었다. 해의 품질을 높이기 위해 선택된 개체에 대하여 지역탐색을 허용하였다. 정확한 개체 평가를 위해서는 최소비용나무를 구해야 한다. 우리는 개체의 평가로 인한 계산 시간을 절감하기 위해서 부모 개체의 나무 비용을 이용한 개체 평가 추정방법을 도입하였다. 실험에 의하면 지역 탐색을 사용하면 해의 품질이 개선되고, 개체 평가 추정을 사용하면 연산 시간이 절약되는 것으로 나타났다. 지역 탐색만을 사용한 진화 해법에 비하여 지역 탐색과 개체 평가 추정을 동시에 사용한 경우



<그림 6> 최적해 및 각 진화해법에서 구해진 해들 간의 절약값 비교



<그림 7> 진화해법 간의 연산시간 비교

해의 품질은 미미하게 나빠졌지만 연산 속도는 25%수준으로 감소하였다.
우리가 제시한 진화 해법과 다른 유전해법결과와 비교한 결과 해의 품질 면에서 더 우수한 것으로 나타났으며, 기존의 우수하다고 알려진 일반적인 휴리스틱과 비교하여도 행의 품질이 우수하였다.

향후 이러한 결과를 이용하여 장애회피 직각거리 스타이너 문제를 해결하려고 한다.

<표 1> Julstrom의 유전 해법과 본 연구의 진화 해법간의 해의 품질 비교

		Julstrom		HE_H		HE_EH		Optimal
문제크기	문제번호	나무비용	증가율(%)	나무비용	증가율(%)	나무비용	증가율(%)	
100	1	7.426	2.396	7.290	0.514	7.283	0.428	7.252
100	2	7.818	3.995	7.550	0.424	7.568	0.669	7.518
100	3	7.431	2.150	7.275	0.001	7.275	0.001	7.275
100	4	7.691	3.454	7.443	0.113	7.469	0.471	7.434
100	5	7.734	2.207	7.577	0.128	7.568	0.007	7.567
100	6	7.725	3.810	7.474	0.439	7.502	0.818	7.441
100	7	8.079	3.923	7.783	0.112	7.792	0.231	7.774
100	8	7.499	2.679	7.315	0.165	7.329	0.352	7.303
100	9	8.003	2.666	7.803	0.100	7.803	0.100	7.795
100	10	7.802	2.722	7.603	0.105	7.609	0.175	7.595
100	11	7.983	1.468	7.884	0.211	7.877	0.127	7.867
100	12	7.760	1.929	7.627	0.177	7.621	0.109	7.613
100	13	7.700	3.210	7.497	0.483	7.497	0.483	7.460
100	14	8.100	3.010	7.888	0.315	7.888	0.315	7.863
100	15	7.270	3.199	7.076	0.450	7.073	0.408	7.045
250	1	11.947	2.453	11.759	0.842	11.735	0.632	11.661
250	2	11.774	2.249	11.577	0.535	11.583	0.593	11.515
250	3	11.750	2.485	11.519	0.471	11.502	0.319	11.465
250	4	12.045	2.233	11.871	0.759	11.864	0.697	11.782
250	5	12.055	3.098	11.770	0.664	11.765	0.614	11.693
250	6	11.934	2.653	11.650	0.206	11.648	0.190	11.626
250	7	11.877	3.030	11.603	0.654	11.610	0.716	11.528
250	8	12.070	3.310	11.798	0.982	11.817	1.147	11.683
250	9	12.052	3.166	11.738	0.475	11.733	0.434	11.682
250	10	12.103	3.570	11.788	0.874	11.764	0.671	11.686
250	11	11.578	2.560	11.358	0.612	11.354	0.575	11.289
250	12	12.227	2.717	11.953	0.413	11.946	0.357	11.904
250	13	11.890	2.456	11.671	0.567	11.711	0.917	11.605
250	14	11.993	3.220	11.687	0.584	11.687	0.584	11.619
250	15	11.870	2.719	11.650	0.813	11.628	0.623	11.556
500	1	16.663	2.240	16.469	1.051	16.472	1.069	16.298
500	2	16.575	3.106	16.213	0.856	16.264	1.171	16.076
500	3	16.830	3.464	16.445	1.099	16.425	0.973	16.266
500	4	16.854	2.699	16.547	0.825	16.539	0.777	16.411
500	5	16.469	2.556	16.172	0.706	16.190	0.819	16.059
500	6	16.864	2.402	16.639	1.034	16.661	1.170	16.469
500	7	16.433	2.627	16.176	1.024	16.200	1.171	16.012
500	8	16.514	2.414	16.283	0.982	16.343	1.352	16.125
500	9	16.630	2.591	16.310	0.619	16.363	0.941	16.210
500	10	16.045	3.129	15.685	0.813	15.711	0.986	15.558
500	11	16.620	2.799	16.302	0.829	16.305	0.852	16.167
500	12	16.822	2.567	16.583	1.109	16.602	1.228	16.401
500	13	16.558	2.638	16.279	0.906	16.296	1.013	16.132
500	14	17.064	2.805	16.795	1.187	16.797	1.198	16.598
500	15	16.574	3.099	16.217	0.878	16.233	0.981	16.076
평균			2.797		0.602		0.655	

<표 2> 최적해, 진화해법, Beasley, Kahng, Borah 과 Julstrom의 해법에 의해 구해진 해들의 절약값 비교 (* : 결과를 제시하지 않은 경우)

n	HE_H	HE_HE	Beasley's	Kahng's	Borah's	Julstrom's	Optimal solution
10	10.611	10.6248	9.947	10.36	10.33	*	10.656
20	11.721	11.7109	10.590	10.44	10.4	*	11.798
30	11.356	11.3057	10.250	*	*	*	11.552
40	10.807	10.7236	9.956	*	*	*	10.913
50	10.705	10.6547	9.522	10.71	10.71	*	10.867
60	11.755	11.7358	10.146	*	*	*	11.862
70	11.163	11.0993	9.779	*	*	*	11.387
80	11.081	11.1423	9.831	*	*	*	11.301
90	11.203	11.1864	10.128	*	*	*	11.457
100	11.501	11.4447	10.139	10.89	10.84	9.203	11.720
250	11.090	11.1128	9.964	10.88	10.88	9.178	11.646
500	10.811	10.7059	9.879	*	10.94	9.208	11.631
mean	11.143	11.1206	10.011	10.656	10.683	9.196	11.399

참고문헌

[1] Back, Thomas, *Evolutionary Algorithm in Theory and Practice*, Oxford University Press, 1996.

[2] Barreiros, J. ,"An Hierarchic Genetic Algorithm for Computing (near) Optimal Euclidean Stein Steiner Trees", *Workshop on Application of hybrid Evolutionary Algorithms to NP-Complete Problems*, Chicago, 2003

[3] Beasley, J. E.," OR-Library: distributing test problems by electronic mail ", *Journal of the Operational Research Society*, Vol.41(1990), pp.1069-1072

[4] Beasley, J. E. " A heuristic for Euclidean and rectilinear Steiner problems", *European Journal of Operational Research*, Vol.58(1992), pp. 284-292

[5] Borah, M., R. M. Owens, " An Edge-Based Heuristic for Steiner Routing", *IEEE Trans. on Computer Aided Design*, Vol.13(1994), pp. 1563-1568

[6] France, R.L., " A note on the optimum location of new machines in existing plant layouts", *J. Industrial Engineering*, Vol.14(1963), pp. 57-59

[7] Ganley, Joseph L., " Computing optimal rectilinear Steiner trees: A survey and experimental evaluation", *Discrete Applied Mathematics*, Vol.90 (1999), pp.161-171

[8] Garey, M. R., D. S. Johnson, " The rectilinear Steiner tree problem is NP-complete", *SIAM Journal on Applied Mathematics*, Vol. 32(1977) pp.826-834

[9] Gottlieb, Jens, B.A. Julstrom, G.R. Raidl, F. Rothlauf, " Prüfer numbers: A poor representation of spanning trees for evolutionary search", *Proceedings of the 2001 Genetic and Evolutionary Computation* (2001), pp. 343-384

[10] Hanan, M., " On Steiner's problem with rectilinear distance", *SLAM Journal on Applied Mathematics*, Vol.14 (1966), pp. 255-265

[11] Hesser, J., R. Manner, O. Stucky, " Optimization of Steiner Trees using Genetic Algorithms", *Proceedings of the Third International Conference on Genetic Algorithm* (1989), pp. 231-236

[12] Ho, J. M., G. Vijayan, C. K. Wong, " New algorithm for the rectilinear Steiner tree problem", *IEEE Trans. on Computer Aided Design* Vol.9(1990), pp. 185-193

[13] Hwang, F. K., " An O(n logn) algorithm for suboptimal rectilinear Steiner trees", *IEEE Transaction son Circuits and Systems*, Vol.26(1979), pp. 75-77

[14] Joseph, J., F. C. Harris, "A Genetic Algorithm for the Steiner Minimal Tree Problem", *Proceedings of International Conference on Intelligent Systems* (1996)

[15] Jin, Y.: A Survey on fitness Approximation in Evolutionary Computation. *Journal of Soft Computing*, Vol.9(2005) pp.3-12

[16] Julstrom, B.A., " A genetic algorithm for the Rectilinear Steiner problem", *Proceedings of the 5th International Conference on Genetic Algorithms* (1993), pp. 474-480

[17] Julstrom, B.A., " Representing

rectilinear Steiner trees in genetic algorithm", *Proceedings of the 1996 ACM Symposium on Applied Computing* (1996), pp. 245-250
[18] Julstrom, B.A., " Encoding Rectilinear Trees as Lists of Edges", *Proceedings of the 16th ACM Symposium on Applied Computing* (2001), pp. 356-360
[19]Julstrom, B.A., " A Scalable Genetic Algorithm for the rectilinear Steiner Problem", *Proceedings of the 2003 Congress on Evolutionary Computation* (2002), pp. 1169-1173
[20] Kahng, A. B., B. Robins, " A New Class of Iterative Steiner Tree Heuristics with Good Performance", *IEEE Trans. on Computer Aided Design,* Vol.11(1992), pp.893-902
[21] Lee, J. L., N. K. Bose, F. K. Hwang, " Use of Steiner's problem in suboptimal routing in rectilinear metric", *IEEE Transaction son Circuits and Systems,* Vol. 23(1976), pp. 470-476
[22] Palmer, C.C., A. Kershenbaum, " Representing trees in genetic algorithms", *Proceedings of the First IEEE conference on Evolutionary Computation,* Vol. 1(1994), pp. 379-384
[23] Soukup, J., W. F. Chow, "Set of test problems for the minimum length connection networks", *ACM/SIGMAP Newsletter,* Vol.15(1973), pp. 48-51
[24] Wakabayashi, S. "A Genetic Algorithm for Generating a Set of Rectilienar Steiner Trees in VLSI Interconnection Layout", *Information processing Society of Japan Journal,* Vol.43, No.5 (2002)
[25] Warme, D.M., P. Winter, M. Zachariasen, " Exact Algorithms for Plane Steiner Tree Problems : A Computational Study " In: D.Z. Du, J.M. Smith and J.H. Rubinstein (eds.): *Advances in Steiner Tree,* Kluser Academic Publishers (1998)
[26] Warme, D. M. , www.group-w-inc.com / ~warme / research
[27] Yang, B.H. "An Evolution Algorithm for the Rectilinear Steiner Tree Problem", *Lecture Notes in Computer Science,* Vol.3483(2005) 241-249
[28] Yang, B.H. " A Nodes Set Based Hybrid Evolutionary Strategy on the Rectilinear Steiner Tree Problem", *Korean Management Science Review,* Vol.23.No.1(2006)75-85
[29] Yang, B.H. " Hybrid Evolutionary Algorithms for the Rectilinear Steiner Tree Problem using Fitness Estimation", *Lecture Notes in Computer Science,* Vol.3982 (2006) 581-589