韓國軍事運營分析 學會誌
第17卷. 第1號. 1991.6.30.

최적화 문제 해결 기법 연구
(Resolutions of NP-complete Optimization Problem)

김동윤, 김상희, 고보연*

Abstract

In this paper, we deal with the TSP (Traveling Salesperson Problem) which is well-known as NP-complete optimization problem. the TSP is applicable to network routing, task allocation or scheduling, and VLSI wiring.

Well known numerical methods such as Newton's Metheod, Gradient Method, Simplex Method can not be applicable to find Global Solution but they just give Local Minimum. Exhaustive search over all cyclic paths requires $\frac{1}{2}$ (n-1) ! paths, so there is no computer to solve more than 15-cities.

Heuristic algorithm, Simulated Annealing, Artificial Neural Net method can be used to get reasonable near-optimum with polynomial execution time on problem size. Therefore, we are able to select the fittest one according to the environment of problem domain.

Three methods are simulated about symmetric TSP with 30 and 50-city samples and are compared by means of the quality of solution and the running time.

* 국방과학연구소

1. 서론

수학, 전산학, 산업공학 등 여러분야에서 연구하고 있는 최적화 문제는 의사결정 지원, 통신망의 routing, 작업할당 및 스케쥴링, 공정제어, VLSI의 배선연결 등 다양하게 응용되고 있다. 본 연구에서는 최적화 문제로서 널리 알려진 TSP(Traveling Salesperson Problem)를 다루어 보았다. TSP문제는 n개 도시들 사이의 거리에 대한 n x n대칭행열이 주어졌을때 모든 도시를 정확히 한번만 방문하는 최단 순환 경로를 찾는 것이다. 이 문제는 NP-complete(Nondeterministic Polynomial) 계열에 속하는 복잡한 최적화 문제로서 exhaustive search에 대하여 현재 가용한 컴퓨터로 15개 도시 이상의 큰 문제에 대하여 최적해(global optimum)를 구하는 일은 거의 불가능하다[1,2]. 따라서 어느정도 신뢰성 있는 근사해(near optimum)를 n에 대한 다항식 함수의 시간 복잡도로 구하는 연구가 활발하다.

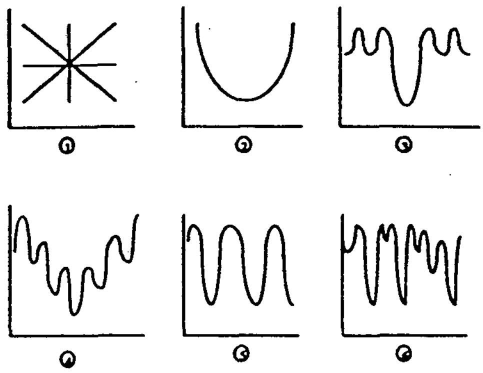
실제 문제에 적용가능한 근사해를 구하는 연구는 크게 heuristic 알고리즘, simulated annealing, 신경회로망(neural network) 방법으로 구분할 수 있다.

Heuristic 알고리즘은 임의의 한 도시부터 시작하여 최단경로를 형성할 수 있도록 다음 도시들을 하나씩 연결하는 경로형성(tour construction) 방법과 임의로 모든 도시를 연결한 초기 경로로부터 부분적으로 연결상태를 조정하여 보다 더 좋은 경로를 구성하는 경로개선(tour improvement) 방법 및 이 둘을 혼합한 방법등 세가지 접근 방법들이 있다[2.3.4.5]. 이 알고리즘들은 85개 도시 이내는 대체적으로 최적해를 얻지만 그 이상에서는 어느정도 신뢰성을 지닌 근사해를 얻는다. 최적해와 근사해의 여러가지 형태는 그림 1과 같다.

Simulated annealing은 불규칙적인 분자 구조를 갖는 물체에 고온으로 열을 가하여 액화상태를 만든다음 열평형상태(thermal equilibrium)를 유지하면서 천천히 온도를 낮추면 양질의 물체로 변환(예를들면, 유리가 수정으로 변환)되는 과정을 컴퓨터 모의시험에 적용하여 Metropolis criterion에 따라 Local minimum을 어느정도 벗어나 보다 더 최적해에 가까운 결과를 얻을 수 있는 방법이다[6].

이 방법을 TSP 문제에 적용하면 초기온도를 높게 설정하여 임의로 초기경로를 형성한 다음 종료조건을 만족할 때까지 온도를 조금씩 낮추면서 보다 더 좋은 경로를 찾아가는, 즉 heuristic 방법중 경로개선 방법과 유사한 처리과정을 수행

한다. 이 방법은 최종해의 정확도를 높일 수는 있지만 상당히 긴 실행시간이 요구된다.



(a) 형 태

번호	최적해 갯 수	근사해 갯 수	형 태
1	없 음	없 음	(1)
2	1 개	없 음	(2)
3	1 개	많 음	(3) (4)
4	많 음	없 음	(5)
5	많 음	많 음	(6)

(b) 분 류

<그림 1 : 최적화 문제의 여러가지 해>

본 연구에서는 실행시간이 긴 단점을 개선하기 위하여 경로형성 방법의 일종인 방위각 연결방법(모든 도시의 median 값을 구하여 각 도시를 median 좌표와 연결한 후 이웃하는 방위각에 해당하는 도시들을 연결하여 경로를 형성)을 이용하여 초기경로를 결정한 다음 annealing을 하였다. 이로써 최종해의 정확도 및 실행시간이 어느정도 향상되었다.

인간 두뇌와 유사한 구조를 가지고 지식을 처리하는 신경회로망 기법은 현재 패턴인식, 최적화문제, 전문가 시스템 등에 활발하게 사용되고 있다[14].

신경회로망 이론을 TSP에 처음으로 적용한 Hopfield/Tank[11] 방법은 n이 약 30정도 까지만 가능하며 최종해의 정확도가 많이 떨어지므로 현재는 탄성망(elasticnet)을 이용한 방법이 주로 연구되고 있다[9,10]. 이 방법은 문제 영역의 중심부분에 작은 원형의 초기 탄성망(노드 갯수는 도시 갯수보다 같거나 많음)을 형성한 다음 학습규칙을 이용하여 탄성망을 점진적으로 확장시켜 탄성망 노드들을 각 도시에 근접시킨후 최단 경로를 형성하는 방법으로 최종해의 정확도는 다소 떨어지지만 실행시간을 크게 단축시킬 수 있다. 신경회로망의 입력층 뉴런에 도시좌표를, 출력층 뉴런에 탄성망 노드를, 각 노드에서 도시까지의 거리를 연결강도 행렬(weight matrix)로 각각 대응시키고 연결강도 행렬이 뉴런간의 경쟁에 의해 조정되는 경쟁학습(competitive learning)을 수행함으로써 최종해를 얻게된다.

각 방법들의 성능을 비교하기 위한 모의시험에는 통상 다음의 4가지 문제 영역들이 사용된다.

첫째, 기존에 발간된 책자나 논문에 제시된 문제

둘째, 2차원 유클리드 공간에 임의로 위치시킨 좌표를 이용

세째, 각 도시간의 거리를 나타낸 거리 행렬을 이용

네째, 실제 문제

본 연구에서는 두번째 문제 영역에 대하여 모의시험을 하였는데, 2차원 유클리드 공간은 단위 사각형안에 uniform 분포로부터 각 점을 독립적으로 발생시켜서 위치시킨 공간을 의미한다. 모의시험에는 30개와 50개 도시에 대하여 각각 10개의 sapmle을 사용하였고, 개선된 simulated annealing과 탄성망 방법을 IBM PC/AT 컴퓨터에 Turbo C 프로그래밍 언어를 이용하여 구현 및 시험하였다.

2. 본 론

일반적인 TSP문제 해결에 이용되는 4가지 방법을 살펴본다.

2.1 Dynamic programming [1]

이 방법은 모든 도시에 대한 상호간의 거리가 비대칭(asymmetric) 거리행렬로 주어졌을때 TSP조건(모든 도시는 경로상에서 반드시 한번만 나타나야 하고, 경로의 시작 도시와 끝 도시를 연결하여 순환경로를 형성해야 함)을 만족하는 모든 가능한 경로들을 구한뒤 최단 경로를 찾는 방법이다. 따라서 이 방법은 언제나 최적해를 구할 수 있지만 처리 과정에 소요되는 기억용량과 실행시간이 $O(n2^n)$과 $O(n^2 2^n)$ 이므로 비 현실적이다.

함수 g(i,s)를 i번째 도시에서 출발하여 집합 s에 있는 모든 도시를 연결한 후 다시 i번째 도시로 돌아오는 최단경로 값이라 정의하면 최적해는 다음 사항을 이용하여 g(1, v-{1})을 구하면 된다.

- $g(1, V-\{1\}) = \min_{2 \leq k \leq n} \{C_{1k} + g(k, V-\{1, k\})\}$
- $g(i, S) = \min_{j \in S} \{C_{ij} + g(j, S-\{j\})\}$ (for $i \notin S$)
- $g(i, \phi) = C_{i1}$, $1 \leq i \leq n$
- V : 문제영역에 있는 n개 도시집합
- 거리행렬 : $[C_{ij}]$

2.2 heuristic 알고리즘

Dynamic programming의 비현실성(큰 값의 n인 경우)에 따라 다항식 함수 형태의 시간 복잡도로서 최적해에 가까운 근사해를 얻을 수 있는 heuristic 알고

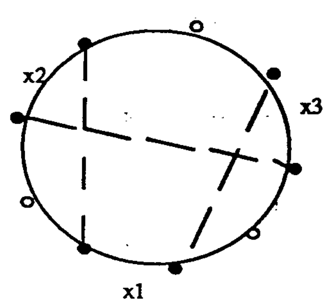
리즘이 연구되었다. 이 부류에 속하는 알고리즘은 크게 3가지 형태로 구분된다.

첫째, 경로 형성 방법

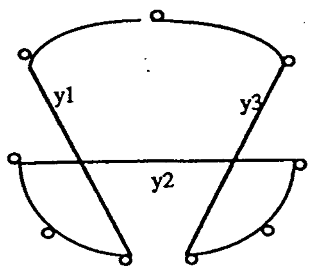
임의의 한 도시부터 시작하여 TSP조건에 맞도록 모든 도시를 연결할 때까지 기존의 경로를 최소로 할 수 있도록 도시 한개씩 연결시켜 가는 방법으로 Nearest neighbor, clarke and wright savings, 5개의 insertion procedures, christofides' heuristics 등이 있다. 이들은 대체로 $0(n^4 \log n)$ 시간 복잡도가 필요하다.

둘째, 경로 개선 방법[2,4,5]

임의의 경로는 최적경로에 비하여 몇 개의 도시간 연결선 위치가 잘못되어 있으므로 이들을 선택해서 최적 위치로 연결시키는 방법이다. 대표적인 알고리즘은 경로를 개선할 때마다 k개의 연결선을 교체하는 "k-opt"알고리즘이다. <그림 2>에는 기존의 연결선 x1,x2,x3를 y1,y2,y3로 교체하는 과정을 나타낸다.



a) 전



b) 후

<그림 2 : 연결선 교체>

일반적으로 k가 클수록 최적해에 더 가까운 해을 얻지만 교체하는 연산이 $0(n^k)$ 이므로 보통 k가 2, 3인 경우가 사용된다.

세째, 조합방법[3]

초기 경로를 경로 형성 방법으로 결정한 후 경로 개선 방법에 따라 최종해를 구한다. 이 방법은 위의 두가지 방법보다 더 최적해에 가까운 근사해를 얻지만 시간이 더욱 길어지는 문제의 해결이 필요하다. 또한 응용문제에 따라 여러가지 대안 중에서 적절한 조합 방법을 결정해야 한다.

2.3 Simulated annealing

Simulated annealing은 Kirkpatrick [6]이 물리학, 열역학에서의 annealing 과정을 최적화 문제에 적용시킨 방법으

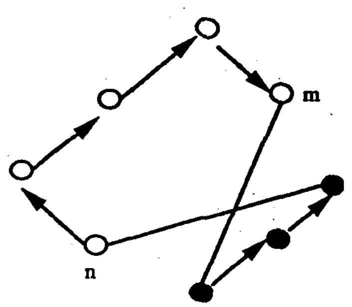
로 Monte Carlo annealing, statistical cooling, probabilistic hill climbing, stochastic relaxation 등으로도 불린다. 이 방법은 임의의 초기 경로에서 시작하여 경로의 연결상태를 변경시켜 비용함수(경로길이)가 감소할 때 뿐만 아니라 증가할 때도 metropolis 규칙을 만족하면 새로운 경로로 바꾸어 줌으로써 어느정도 근사해에서 벗어나 최종 결과가 최적해에 상당히 근접할 수 있으며 이의 수행에는 다음과 같은 4가지 정의가 필요하다.

첫째, 초기 경로 상태(configuration)

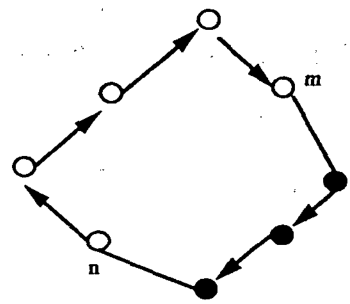
n개의 도시좌표 (x_i, y_i)에 대하여 1부터 n까지의 순열 중 임으로 한개를 선택한다.

둘째, 경로 변경(rearrangement)

경로 중 일부분을 선택하여 연결순서를 반대 방향으로 바꾸는 경로역전 (<그림 3>)과 같이 임의의 2개 도시 사이로 옮기는 경로이전 (<그림 4>)으로 변경한다.

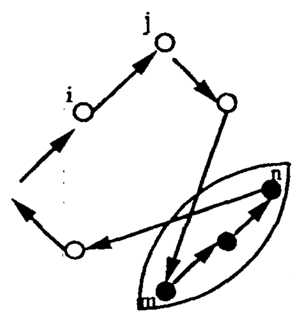


a) 전

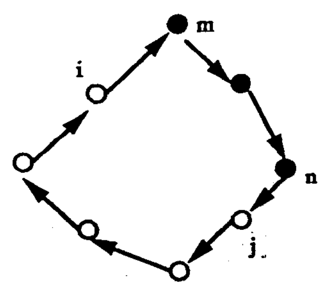


b) 후

<그림 3 : 경로역전(reverse)>



a) 전



b) 후

<그림 4 : 경로이전(transport)>

셋째. 비용함수

경로상에 있는 각 도시간 거리의 합을 비용으로 결정한다.

$$E=\sum_{i=1}^{n}\sqrt{(x_i-x_{i+1})^2+(y_i-y_{i+1})^2}$$
$$(n+1\equiv 1)$$

넷째. annealing 계획 (cooling 계획)

이 과정은 simulated annealing 과정에서 가장 중요하며, 다음 4가지 매개변수에 따라 최종해의 정확도 및 실행시간이 결정된다.

o 초기 온도

o 감온 비율 (temperature decreasing rate)

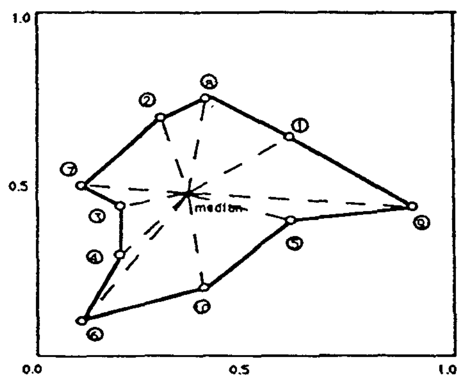
o Markov chain 길이 : 임의의 온도에서 가능한 최대 경로 변경 회수

o 종료 조건 : 더 이상의 경로 변경이 없거나 특정 온도에 도달했을 경우

한편 초기 경로를 임의로 연결한 경로가 아닌 어느정도 잘 정리된 경로로 선택하면 불필요한 시간을 줄일 수 있을 뿐만 아니라 〔3〕의 모의시험 결과와 같이 보다 더 좋은 해를 얻을 수 있다.

본 연구에서는 방위각 연결방법 (azimuthal angle method)에 의한 초기 경로를 제안하였으며 모의 시험 결과 50개 도시 이하에서 약 50%이상 초기 경로 길이를 단축시켰다.

방위각 연결방법은 모든 도시들의 median 좌표를 구하고 이 값을 중심으로 하는 단위 원을 설정한 다음 모든 도시와 원의 중심을 가상적으로 연결하여 원주상에 대응시킨 후 중심과 이루는 방위각에 따라 이웃한 도시를 실제로 연결하여 경로를 형성한다. 〈그림 5〉는 10개 도시에 대한 방위각 연결방법을 보여준다.



〈그림 5 : 방위각 연결방법을 이용한 경로 형성〉

본 연구에서 적용한 simulated annealing 알고리즘은 다음과 같다.

단계1 : 방위각 연결방법에 의한 초기 경로 결정 (P_{old})

단계2 : $T=T_0$ (T_0 : 초기온도), $P=P_{old}$

단계3 : repeat

repeat

P_{new} ← perturb P_{old} by path arrangement

$\triangle E=E(P_{new})-E(P_{old})$

$$\text{if } \Delta E < 0 \text{ then } P \leftarrow P_{new}$$
$$\text{else } P \leftarrow P_{new} \text{ with } \exp(-\Delta E/T) > \varepsilon \quad (0 < \varepsilon < 1)$$
until equilibrium
$$T = \alpha \cdot T \qquad (0 < \alpha < 1)$$
$$P_{old} \leftarrow P$$
until stop criterion

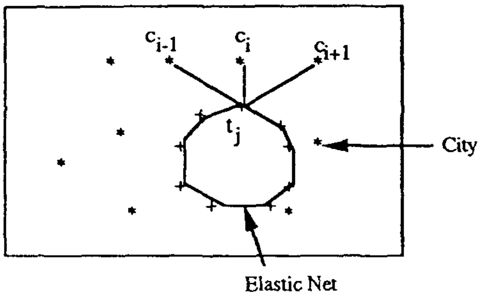
2.4 신경회로망을 이용한 탄성망 (elastic network) 방법

최근 인공 신경회로망의 연구가 활발해지면서 최적화 문제를 신경망 기법으로 해결하는 새로운 접근방법이 제안되었다[9-12]. 신경회로망을 최적화 문제에 적용한 대표적인 예는 Hopfield/Tank의 소위 Hopfield 신경망 모델을 이용한 TSP 해결방안 즉, TSP 조건을 에너지 함수(Lyapunov)로 정의하고 이를 최적화시키는 방법이며 n^2의 뉴런으로 구성된 Hopfield모델을 이용하여 최종적으로 이들 뉴런의 상태가 안정될때까지 계속적으로 변화시키는 과정을 반복한다. Hopfield 모델에서 각 뉴런은 시간에 따르는 상태방정식에 따라 값이 변화되며 에너지함수를 점진적으로 최소화시킨다[11]. 이 방법을 10개, 30개 도시에 대해서 모의 시험한 결과 최적해보다 20% 정도 벗어나고 도시수가 많아짐에 따라 성능이 현저히 저하되어 50개 도시 정도면 TSP 해결방법으로 부적절하다고 판명되었다.

또 다른 신경회로망 기법을 이용한 TSP 해결방안으로 탄성망(elastic net)을 이용한 방법이 제안되었다. 이 방법은 TSP 문제가 유클리드 공간상에서 정의되어 비용함수가 각 도시간의 위치에 대한 거리로 주어질때 사용 가능하다. 이는 Kohonen의 Self-organizing Feature Map(KSFM)과 유사한 구조로 경쟁학습을 통하여 문제를 해결한다. 탄성망 방법은 Burr에 의해서 제안되었고 이를 Durbin & Wilshaw가 TSP 문제에 적용하여 1% 오차 이내의 근사해를 얻었다. 그러나 이 방법은 1000회 이상의 반복학습이 필요하며 많은 시간을 소비하는 단점이 있다. 본 연구에서는 Durbin & Wilshaw의 방법을 개선하여 학습회수를 감소시킨 개선된 탄성망(improved elastic net) 방법을 사용하였다[10].

〈그림 6〉은 유클리드 공간상에 퍼져있는 도시(C_i)와 그 중심부에 위치한 탄성망(t_j)을 나타낸다. n개의 도시에 대한 탄성망 노드수 m은 m>=n이면 가능하며 초기에 형성된 각 탄성망 노드들은 여러곳에 위치한 모든 도시들과의 거리에 따라서 노드의 좌표값이 변화(학습)되므로 탄성망의 형태가 변화된다. 이때 경로상

의 연계들이 서로 교차되지 않도록 탄성망 노드들 간의 이웃 관계를 잘 유지해야 한다.



〈그림 6 : 도시좌표와 초기 탄성망〉

탄성망을 변형시키는 힘은 다음 3가지로 구분할 수 있다.

1. $F_i = \alpha \sum_{i}^{n} W_{ij}(C_i - t_i)$: Foward Force

2. $B_i = \alpha \sum_{i}^{n} V_{ij}(C_i - t(c_i))$ Backward Force

3. $T_i = \beta K(t_{i+1} - 2t_j + t_{j-1})$: Neighbours Force

(α, β : 0-1인 상수, $t(c_i)$: 도시 c_i에 가장 가까운 탄성망 노드)

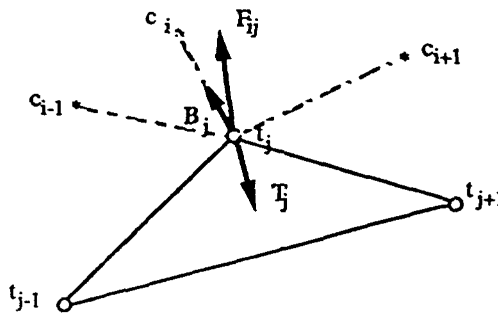
첫째항 F_i는 모든 도시(c_i)와 탄성망 노드(t_i)와의 거리에 Gaussian weight w_{ij}를 곱하여 t_i의 좌표를 변화시키므로 도시가 밀집된 지역으로 이동, 확장케 한다. 둘째항 B_i는 탄성망의 각 노드와 실제로 가까운 도시 사이에 작용되는 힘으로 D & W의 탄성망 모델에 새로이 추가시킨 항이다. 세째항 T_i는 노드들 사이에 작용되는 힘으로 전체적인 망의 형태를 둥글게하여 경로길이가 최소가 되도록 도움을 주며 탄성망이 교차되지 않도록 하는 힘이다. 여기서 각 Gaussian weight w_{ij} v_{ij}는 아래식으로 정의되며 K는 simulated annealing에서 온도변수에 대한 역할과 같으며 수렴속도를 높이기 위하여 매 학습마다 각 도시와 가장 가까운 탄성망 노드 사이의 거리를 rms(root mean square) 값으로 취한다.

$$w_{ij} = \Phi(|c_i - t_i|, K) / \sum \Phi(|c_i - t_i|, K)$$

$$V_{ij} = \Phi(|t_i - t(c_i)|, K) / \sum \Phi(|t_i - t(c_i)|, K)$$

$$K = [\sum_{1} |c_i - t(c_i)|^2 / N]^{1/2}$$

$$(\Phi(d, k) \equiv \exp(-d^2/k^2))$$



〈그림7 : 탄성망에 작용하는 3종류의 힘〉

위에서 언급한 이들 세 힘의 작용을 도시하면 〈그림 7〉과 같으며 이때 각 탄성망 노드의 좌표변화(학습)는 다음 식으로 주어진다.

$$\triangle t_i = F_i + B_i + T_i$$

최종적으로 각 탄성망 노드가 어느 한 도시의 좌표에 충분히 근접할때까지 반복 학습시킨다. 이 과정을 통하여 초기 탄성망 노드간에 연결된 경로가 변하여 결국에는 최종해가 됨을 알 수 있다. 이 방법은 기존의 방법과 비교해서 실행시간이 짧으며 학습률(α)과 k매개변수에 대해서 비교적 안정된 특성(얻은 해가 크게 변하지 않음)을 보인다. 또한 각 탄성망 노드들은 동일한 기능을 수행하므로 이들을 독립된 processor로 할당하여 동시에 처리할 수 있는데 비록 이들이 학습규칙상에서 완전 독립된 형태는 아니더라도 대부분의 연산이 도시와 탄성망노드 혹은 탄성망노드 간의 거리계산에 의존하므로 많은 부분을 병렬처리로 구현할 수 있는 잇점이 있다. 그러나 이 방법 역시 도시수가 많아짐에 따라 최적해에서 보다 멀어지므로 학습률(α)을 낮추고 k(온도)를 서서히 감소시켜야 보다 좋은 결과를 얻을 수 있다.

3. 모의 시험 결과

모의 시험은 30개, 50개 도시에 대하여 각각 10set의 sample을 2차원 유클리드 공간에서 추출하여 IBM PC/AT에서 수행하였다. 본론에서 살펴본 4가지 방법 중 실행시간, 기억용량, 정밀도에서 현실 문제에 적용 가능한 simulated annealing과 탄성망을 이용한 신경회로망 방법을 모의 시험하였다. 〈표 1〉에는 이 두가지 방법과 최적해를 경로길이와 실행시간으로 비교하였다.

〈표 1 : simulated annealing과 탄성망 방법 비교〉

sample	simulated annealing		탄 성 망		최적해
	길 이	시 간 (sec)	길 이	시 간 (sec)	길 이
#1	4.53	154.0	4.60	75.0	4.53
#2	4.02	137.0	4.02	61.0	4.02
#3	4.65	154.0	4.96	72.0	4.65
#4	4.77	161.0	4.77	66.0	4.77
#5	4.66	126.0	4.66	61.0	4.66
#6	3.92	119.0	4.00	66.0	3.92
#7	4.64	170.0	4.67	55.0	4.63
#8	4.46	147.0	4.63	61.0	4.46
#9	4.29	127.0	4.36	55.0	4.29
#10	4.60	188.0	5.00	77.0	4.60
평균	4.45	148.3	4.57	64.9	4.45

(a) 30개 도시

sam-ple	simulated annealing		탄 성 망		최적해
	길 이	시 간 (sec)	길 이	시 간 (sec)	길 이
#1	5.98	227.0	6.17	221.0	5.97
#2	5.44	228.0	5.67	172.0	5.44
#3	5.60	252.0	5.81	249.0	5.57
#4	5.67	228.0	5.87	280.0	5.67
#5	5.39	203.0	5.50	252.0	5.25
#6	6.26	185.0	6.20	266.0	6.07
#7	5.54	277.0	5.66	239.0	5.52
#8	5.42	240.0	5.66	252.0	5.41
#9	5.97	353.0	6.19	221.0	5.84
#10	5.67	294.0	5.84	207.0	5.67
평균	5.69	254.7	5.86	235.9	5.64

(b) 50개 도시

최적해는 모든 가능한 경로를 고려하여 결정해야 하지만 계산이 불가능하므로 simulated annealing과 탄성망 방법에서 매개변수 값을 다양하게 설정한 후 각 sample에 대하여 얻은 최소 경로 길이를 최적해로 결정하였다. simulated annealing과 탄성망 방법의 해는 10개의 sample에 대하여 평균값이 최소가 되는 특정한 매개변수 값에서의 결과를 나타낸 것이다. 최종해의 정밀도는 위의 두가지 방법에 대하여 30개 도시에서 (0%, 1%), 50개 도시에서 (3%,4%) 정도의 오차가 있다.

Simulated annealing 방법의 실행시간은 모의시험에 사용한 sample뿐만 아니라 임의로 선정한 80개, 100개 도시에 대해서도 $n\log_2 n$ ($30\log_2 30 \fallingdotseq 147$, $50\log_2 50 \fallingdotseq 282$) 이내에 수행된다. 탄성망 방법을 이용할 경우 상용 컴퓨터로는 실행시간에서 잇점이 별로 없지만 neuro-computer 상에서 구현시 시간복잡도 (기존의 시간 복잡도÷뉴런의 갯수)를 감소시킬 수 있으므로 큰 영역의 문제에도 적용 가능하다.

Simulated annealing 방법에서 annealing 계획은 [7,8]에 제시된 여러가지 매개변수에 대한 수학적 모델 중에서 가장 단순한 방법을 선택하였다. 온도 감소는 일정비율(10%,9.25%)에 따라 선형적으로 감소시켰고 Markov chain 길이는 모든 온도에서 동일하도록 결정하였으며 더 이상의 경로개선이 없을 때 실행을 종료하도록 하였으며, 30개와 50개 도시에 대하여 각각 평균적으로 40, 42 iterations를 수행하였다. 탄성망 방법에서는 $\alpha=0.2$, $\beta=1.0$일때 30개와 50개 도시에 대하여 각각 평균적으로 12, 16 iterations를 수행하였다.

4. 결론 및 연구방향

의사결정 지원 등 여러 분야의 최적화 문제 해결 방안으로 응용 가능한 TSP (Traveling Salesperson Problem) 문제에 대하여 대표적인 4가지 방법을 설명하였으며, 특히 대규모 문제 영역에 대하여 최종해의 정밀도와 실행시간에서 현실적인 simulated annealing과 신경회로망을 이용한 탄성망(elastic network) 방법을 집중적으로 다루었다.

Simulated annealing은 Metroplis 규칙(exp(-△E T) ε)에 따라 local minimum에서 어느정도 벗어나 최적해에 근접한 결과를 얻을 수 있다. 이 방법에서는 초기온도, 감온비율, Markov chain 길이, 종료조건 등 4가지 매개변수를 결정하는 annealing 계획이 중요하다. 감온비율을 비선형 함수(log 함수의 역함수 형태)로 결정하면 최종해의 정밀도는 증가하지만 낮은 온도에서의 불필요한 경로변경으로 실행시간이 너무 길어진다. 따라서 일반적으로 선형함수(일정비율로 온도를 감소시킴)를 사용한다.

신경회로망 구조 및 학습규칙을 이용하여 문제를 해결하는 탄성망 방법은 문제영역의 중심부에 탄성망을 형성한 후 탄성망 노드를 도시좌표에 점차 근접시키는 경쟁 학습규칙에 따라 최종해를 얻는다. 이 방법은 Forward, Backward, Neighbor 등 3가지 힘에 따라 탄성망이 변형되는데 최종해의 정밀도는 다소 떨어지지만 빠른 시간내에 최종해를 얻을 수 있으며, 특히 neurocomputer(고밀도 분산형 병렬구조)가 개발되면 신경회로망이 갖는 병렬성을 이용하여 대부분의 연산(도시와 탄성망노드, 탄성망 노드간 계산)을 병렬로 처리할 수 있게 된다.

본 연구에서는 비용함수가 도시간 거리로만 나타난 경우를 다루었으나 현실문제에 적용하기 위해서는 각 도시에서의 비용이 추가되고 모든 도시간 거리에 weight factor를 고려한 보다 더 복잡한 TSP를 해결할 수 있는 방안이 연구되어야 하며, 더욱 복잡한 문제의 해결을 위하여는 지식표현, heuristic을 이용한 전문가 시스템의 개발 더 나아가서는 불확정성의 주관적 판단 및 fuzzy set 이용 등의 연구가 뒤따라야 할 것이다.

5. 참고문헌

1) E.Horowitz and S.Sahni, *Fundamentals of Computer Algorithms,* 1978, computer science press, pp. 231-412.
2) S.Lin and B.W.Kernighan, "An Effective Heuristic Algorithm for the TSP," *Operations Research 21,* 1973, pp.498-516.
3) B.Golden, et al., "Approximate Traveling Salesman Algorithms," *Operations Research 28,* 1980, pp.694-711.
4) E.L.Lawler and D.E.Wood, "Branch and Bound methods : A Survey," *Operations Research 14,* 1966, pp.699-719.
5) R.Jonker and T.Volgenant, "Nonoptimal edges for the symmetric TSP," *Operations Reserch* 32, 1984, pp.837-846.
6) S.Kirkpatrick, et al., "Optimization by Simulated Annealing," *science* 220, 1983, pp.671-680.
7) B.Hajek, "Cooling schedules for optimal annealing," *Mathematics of Operations Research,* 1986, pp.311-329.
8) P.J.M. van Laarhoven and E.H.L.Aarts, "*Simulated Annealing : Theory and Applications,*" 1987, A Member of the Kluwer Academic Publishers Group, pp.55-98.
9) R.Durbin and D.Willshaw, "An analogue approach to the TSP using an elastic net method," *Nature* 326, 1987, pp.689-691.
10) D.J.Burr, "An Improved Elastic Net Method for the TSP," *IEEE ICNN,* 1988, pp.69-76.
11) J.J.Hopfield and D.W.Tank, "Neural Computation of Decisions in Optimization Problems," *Biological Cybernetics* 52, 1985, pp.141-152.
12) G.W.Davis, Jr., "Sensitive Analysis in Neural Net Solutions," *IEEE Tr. on Sys. Man and Cyb.*, Vol. 19, No.5, 1989, pp.1078-1082.
13) T.Kohonen, *Self-Organization and Associative Memory,* 1984, Springer-Verlag.
14) D.E.Rumelhart, et al., *Parallel Distributed Processing,* 1986, Vol. I, The MIT Press.