

비대칭 외판원문제에서 호의 후보집합 결정

김헌태* · 권상호* · 지영근* · 강맹규*

Determination of Arc Candidate Set for the Asymmetric Traveling Salesman Problem

Hun-Tae Kim* · Sang-Ho Kwon* · Young-Gun G* · Maing-Kyu Kang*

■ Abstract ■

The traveling salesman problem (TSP) is an NP-hard problem. As the number of nodes increases, it takes a lot of time to find an optimal solution. Instead of considering all arcs, if we select and consider only some arcs more likely to be included in an optimal solution, we can find efficiently an optimal solution. Arc candidate set is a group of some good arcs. For the Lack of study in the asymmetric TSP, it needs to research arc candidate set for the asymmetric TSP systematically. In this paper, we suggest a regression function determining arc candidate set for the asymmetric TSP. We established the function based on 2100 experiments, and we proved the goodness of fit for the model through various 787problems. The result showed that the optimal solutions obtained from our arc candidate set are equal to the ones of original problems. We expect that this function would be very useful to reduce the complexity of TSP.

Keyword : Asymmetric Traveling Salesman Problem, Arc Candidate Set

1. 서 론

외판원문제(TSP : traveling salesman problem)
는 교점집합 $N = \{1, 2, \dots, n\}$, 호집합 $A = \{(i, j) :$

$i \neq j, \forall i, j \in N\}$, 호 (i, j) 의 비용이 c_{ij} 인 네트워크 $G(N, A)$ 에서 시작교점을 출발하여 n 개의 모든 교점을 반드시 한 번만 방문하고 시작교점으로 돌아오는 최소비용의 해밀턴순환로(Hamiltonian cy-

논문접수일 : 2003년 1월 31일 논문게재확정일 : 2003년 5월 29일

* 한양대학교 산업공학과, 연락저자 : 권상호(ksha@logistex.hanyang.ac.kr)

cle)를 찾는 문제이다[1]. 외판원문제는 두 교점간 호의 비용이 양방향으로 모두 같은, 즉 $c_{ij} = c_{ji}$ 인 대칭 외판원문제(symmetric TSP)와 그렇지 않은 비대칭 외판원문제(ATSP : asymmetric TSP)로 분류된다. 외판원문제는 경로결정 문제, 배달 문제, 스케줄링 문제, 스택커 크레인 문제등 다양한 형태의 현실문제에서 그 응용분야를 찾을 수 있다[9]. 이러한 폭 넓은 응용성에도 불구하고 외판원문제는 NP-hard로서 현실적인 크기의 문제는 최적해가 거의 불가능하다[7].

외판원문제에 대한 해법은 최적화 해법(optimization method)과 발견적 해법(heuristic method)으로 구분할 수 있다. 최적화 해법은 최적해를 보장하는 해법으로 분지한계법(branch & bound)과 동적계획법(dynamic programming)등이 있지만 다항식(polynomial) 계산량(complexity)을 갖는 최적화 해법은 발견되지 않아 최악의 경우 계산량이 지수적으로 증가한다. 이러한 이유로 최적해는 보장하지 못하지만 빠른 시간에 최적해 또는 최적해에 근사한 해를 구하는 발견적 해법에 대한 많은 연구가 이루어져 왔다. 발견적 해법은 경로구성방법과 경로개선방법으로 구분할 수 있다. 경로구성방법에는 최근거리인접점(nearest neighbor), 삽입(insertion), Christofides, Savings, Patching 알고리즘 등이 있으며, 경로개선방법에는 k -opt, Lin-Kernighan 알고리즘 등이 잘 알려져 있다[5, 8]. 그러나 비록 효율적인 발견적 해법이라 하더라도 작은 문제에서는 빠른 시간에 최적해 또는 최적해에 근사한 해를 찾지만, 문제가 커지면, 예를 들어 $n = 10^3 \sim 10^5$ 이면 고려해야 할 호의 개수가 n^2 , 즉 $|A| = 10^6 \sim 10^{10}$ 이 되어 빠른 시간에 해를 구하기 어렵다. 따라서 효율적인 계산량으로 최적해 또는 최적해에 근사한 해를 구하기 위해서는 집합 A 의 모든 호를 고려하지 않고 최적 순환로에 포함될 가능성이 높은 일부 호들, 즉 호의 후보집합(arc candidate set)만을 고려할 필요가 있다.

Sakakibra[10]는 최단경로를 구성하는데 고려할 필요가 없는 호들의 특성을 찾았다. Hidaka[4]는

호의 후보집합의 크기가 문제의 해를 구할 때 CPU time, 자료 구조의 크기 그리고 해의 질에 영향을 미친다고 하였다. Junger *et al.*[6]은 비용이 큰 호를 제거한 호의 후보집합이 외판원문제를 위한 다양한 발견적해법에서 적용되고 있음을 보여주었다. Reinelt[8]는 외판원문제와 관련된 최근 연구들을 호의 후보집합을 사용하여 기존의 해법들과 비교 분석하였다. 이러한 연구에도 불구하고 호의 후보집합은 대칭 외판원문제에 한정되어 연구되었다[6, 8]. 본 연구에서는 모든 교점들이 완전 연결된 비대칭 외판원문제에서 교점의 개수에 따라 각 교점에 연결된 호들 중 고려해야 할 호들을 결정하여 호의 후보집합을 구하는 회귀함수를 제시한다. 이를 위해 총 2100개 문제의 최적 순환로를 분석하여 회귀함수를 구하고, TSPLIB[9]문제들을 포함하여 총 787개 문제에 대해 회귀함수를 검증한다.

2. 외판원문제와 후보집합

네트워크 $G(N, A)$ 에서 외판원문제를 수리적으로 모형화하면 다음과 같다.

$$\begin{aligned} \text{minimize } z &= \sum_{i \in N} \sum_{j \in N} c_{ij} x_{ij} \\ \text{subject to} \\ \sum_{i=1}^N x_{ij} &= 1, \forall j \in N \end{aligned} \quad (1)$$

$$\sum_{j=1}^N x_{ij} = 1, \forall i \in N \quad (2)$$

$$x_{ij} = 0 \text{ 또는 } 1, \forall (i, j) \in A \quad (3)$$

$$\sum_{i \in S} \sum_{j \in N-S} x_{ij} \geq 1, \forall S, |S| \geq 2 \quad (4)$$

여기서 변수 x_{ij} 가 1이면 호 (i, j) 가 해밀턴순환로에 포함되며, 0이면 그렇지 않다. 식 (1)은 각 교점 j 로 들어오는 호는 하나뿐임을 의미하고 식 (2)는 각 교점 j 에서 나가는 호는 하나뿐임을 의미한다. 식 (4)는 해밀턴순환로 조건으로 S 는 임의의 교점집합이며, 집합 S 와 $N-S$ 사이를 적어도

한 개 이상의 호로 연결되어야 함을 의미한다.

대칭 외판원문제에서 호의 후보집합에 대한 연구는 최근접교점(nearest neighbors), Delaunay, 그리고 이들을 결합한 방법으로 분류할 수 있다[6, 8]. 최근접교점 방법은 각 교점들에 대해 k 번째까지 가까운 교점들을 연결하는 호들을 모두 호의 후보집합에 포함시킨다. Bentley[2]와 Johnson[5]는 각 교점들에 대해 k 를 약 5~20에서 결정하였다. 그런데 이러한 호의 후보집합은 적은 개수의 호들로 구성되어 계산시간이 빠르다는 장점이 있지만, 교점들이 집단을 이루는 클러스터(cluster)가 존재할 경우 클러스터와 클러스터를 연결하는 호들을 호의 후보집합에 포함시키는 것을 보장하지 못하여 해를 만들지 못하는 단점이 있다.

Delaunay 방법은 Voronoi diagram의 Delaunay graph를 기반으로 하여 구성된다[8]. Delaunay 후보집합은 최근접교점 후보집합보다 많은 호들을 포함하여 클러스터들을 서로 연결하는 호들을 후보집합에 포함시키는 장점이 있다. 그러나, 호의 후보집합을 구성하는데 많은 시간이 걸리는 단점이 있다.

최근접교점과 Delaunay를 결합한 방법은 두 후보집합의 장점들을 갖는다[6]. 이 후보집합은 적은 개수의 호들을 가지고 호의 후보집합을 구성하는 최근접교점 후보집합과 클러스터와 클러스터를 연결하는 호를 포함시키는 Delaunay 후보집합의 장점들을 특성으로 한다.

3. 제안하는 호의 후보집합

호의 후보집합만을 고려하여 최적에 가능한 한 근접한 순환로를 구하기 위해서는 호의 후보집합에 최적 순환로를 구성하는 호가 포함되어야 한다. 이를 위해 본 연구에서는 다양한 비대칭 외판원문제의 최적 순환로를 구성하는 모든 호를 분석하고, 이 호들을 포함하는 호의 후보집합을 구한다. 기호를 다음과 같이 정의한다.

- A_{opt} : 최적 순환로를 구성하는 호의 집합
- $O(i, j)$: 교점 i 에서 출발하는 모든 호들 중 c_{ij} 의 오름차순 순위
- $P(i, j)$: 교점 j 로 도착하는 모든 호들 중 c_{ij} 의 오름차순 순위

여기서 동일 비용의 호들은 동일한 순위를 갖는다. 호(i, j)에 대해 $O(i, j)$ 와 $P(i, j)$ 의 최소값은 호(i, j)의 최소순위인데 이를 $MO(i, j)$ 라 하고 다음과 같이 정의한다.

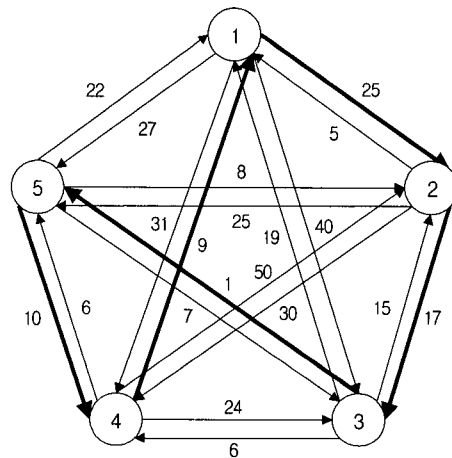
$$MO(i, j) = \min\{O(i, j), P(i, j)\}, \quad \forall (i, j) \in A_{opt}$$

최적순환로를 구성하는 모든 호들에 대해 $MO(i, j)$ 를 구한 후 이들의 최대값, 즉 Max-Min 순위를 MMO라 하고 이를 다음과 같이 정의한다.

$$MMO = \max\{MO(i, j) : \forall (i, j) \in A_{opt}\}$$

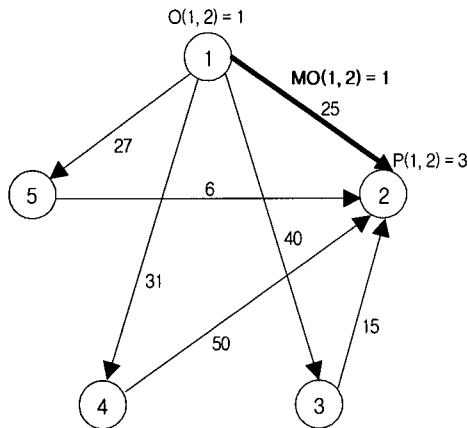
호의 후보집합은 각 교점에서 출발하는 호들 중 비용이 MMO 안에 드는 모든 호들과 각 교점에 도착하는 호들 중 비용이 MMO 안에 드는 모든 호들을 포함한다.

[그림 1]의 $n=5$ 인 예를 통해 이를 설명한다. 그림에서 굵은 화살표는 최적 순환로를 구성하는 호이며, 화살표 옆의 숫자는 각 호의 비용이다.



[그림 1] 비대칭 외판원문제의 예

호(1, 2)를 고려해 보자. 교점 1에서 출발하는 호들 중 c_{12} 의 으뜸차순 순위는 1이다(즉, $O(1, 2)$ 는 1). 또한 교점 2에 도착하는 호들 중 c_{12} 의 으뜸차순 순위는 3이다(즉, $P(1, 2)$ 은 3). 따라서, 호(1, 2)의 최소순위인 $MO(1, 2)$ 는 1이다. [그림 2]는 $O(1, 2)$, $P(1, 2)$, 그리고 $MO(1, 2)$ 를 나타낸 그림이다.



[그림 2] 호(1, 2)의 최소순위

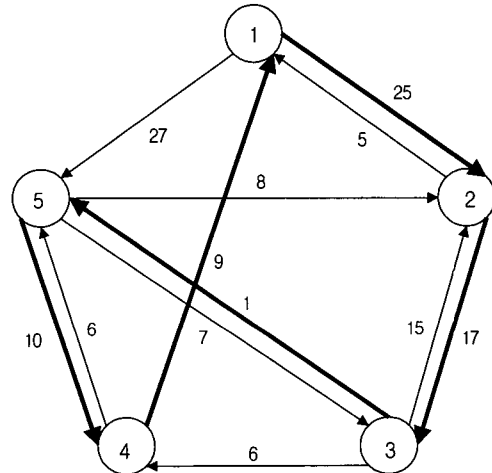
같은 방법으로 최적 순환로를 구성하는 모든 호들에 대해 각 호(i, j)의 $MO(i, j)$ 를 구한 후 MMO 를 구한다. 이 예제에서 MMO 는 2이며, 이는 <표 1>과 같다.

<표 1> MMO 결정

호			순 위		
i	j	c_{ij}	$O(i, j)$	$P(i, j)$	$MO(i, j)$
1	2	25	1	3	1
2	3	17	2	2	2
3	5	1	1	1	1
5	4	10	3	2	2
4	1	9	2	2	2
MMO					2

MMO 로부터 각 교점에서 출발하는 호들 중 비용이 MMO 안에 드는 모든 호들과, 각 교점에 도착하는 호들 중 비용이 MMO 안에 드는 모든 호

들로 호의 후보집합을 구성한다. 이렇게 구한 호의 후보집합을 그림으로 표현하면 [그림 3]과 같다. 이 예에서는 MMO 가 2이므로 각 교점에서 출발하는 호들 중 비용이 순위 2안에 드는 모든 호들과 각 교점에 도착하는 호들 중 비용이 순위 2안에 드는 모든 호들을 호의 후보집합으로 포함한다. 중복되어 포함되는 호들로 인하여 실제 호의 후보집합에 포함되는 호의 개수는 [그림 3]과 같이 12개이다. 이 호의 후보집합안에는 최적 순환로를 구성하는 호가 모두 포함되며 이 호들만을 고려하여 최적 순환로를 구할 수 있다.



[그림 3] 호의 후보집합

4. 실험 및 회귀함수

MMO 값을 이용하면 [그림 3]의 예와 같이 최적 순환로를 구성하는 모든 호를 포함하는 호의 후보집합을 구할 수 있다. 그러나 MMO 값을 구하기 위해서는 문제의 최적 순환로를 구성하는 각 호를 분석하여 $MO(i, j)$ 를 구해야 한다. 본 연구는 이러한 MMO 값을 구하기 위해 문제의 크기, 즉 교점개수 n 만을 이용한 회귀함수를 제시한다. k 를 MMO 를 대신하는 값이라고 하자. k 값을 결정하기 위해서는 다양한 비대칭 외판원문제에서 최적 순환로를 구성하는 호들을 분석하여 MMO 값들을 구하고, 이

값들에 대해 $\bar{x} + 3s$ 를 구하여 회귀함수를 도출한다. 여기서 \bar{x} 는 MMO값들의 모평균이고 s^2 는 표본분산이다. 본 연구에서 제시하는 회귀함수모형은 다음과 같다.

$$k_i = a + b n_i + \varepsilon_i \quad i = 1, 2, \dots \quad (5)$$

- z_i : 교점의 개수
 - n_i : 고려해야 할 호의 최소비용의 순위
 - a : 절편(회귀계수)
 - b : 기울기(회귀계수)
 - $\varepsilon_i \sim iid N(0, \sigma^2)$
- 여기서 a, b 는 미지의 상수이다.

4.1 실험

문제의 크기에 적합한 후보집합을 결정하는 회귀함수를 구하기 위해 다양한 크기의 총 2100개의 비대칭 외판원문제들로부터 MMO 값들을 구하는 실험을 하였다. 실험을 위해 호의 비용이 일양분포와 정규분포를 따르는 두 가지 유형의 비대칭 외판원문제들을 랜덤하게 생성하였다.

호의 비용이 일양분포를 따르는 비대칭 외판원문제를 교점 개수만큼의 비용범위 내에서 랜덤하게 발생시켰다. 예를 들어 교점 개수가 100개인 문제들은 모든 호의 비용이 1과 100사이에서 일양분포를 따르도록 생성하였다. 교점개수가 10개에서 100개까지는 10단위로 크기를 나누었으며, 100개부터 300개까지는 50단위로 크기를 나누어 각 100개씩 문제를 생성하여 실험하였다. 즉, 호의 비용이 일양분포를 따르는 총 1400개 문제를 생성하여 실험하였다. 최적알고리즘을 사용해서 모든 문제의 최적 순환로를 구하고, 1400개의 MMO 값들을 구했다. 각 문제 크기에 해당하는 100개의 MMO값들의 최대값(max), 표본평균값(\bar{X}), 표본표준편차(s)를 계산하였다. 실험을 통해 얻은 일양분포 문제들의 MMO 통계치는 <표 2>와 같다.

호의 비용이 정규분포를 따르는 문제들은 다음과 같은 방법으로 생성하였다. 먼저 일정 범위를

갖는 x, y 좌표에서 교점 개수만큼의 좌표데이터를 랜덤하게 발생시킨다. 그 다음 각 좌표데이터를 기준으로 나머지 좌표데이터들과의 거리를 구한다. 마지막으로 거리의 $\pm 20\%$ 내에서 값에 변화를 주어 비대칭 외판원문제를 생성한다. 예를 들어서 교점 개수가 100개인 문제들은 0과 100사이에서 x, y 좌표 100쌍을 랜덤하게 발생시킨 후 각 좌표간의 모든 거리를 구하고 값에 변화를 주어 비대칭 외판원문제를 발생시킨다. 이런 방식으로 생성된 호의 비용 데이터들은 Arena 3.5의 Input Analyzer를 이용하여 분포를 추정해 본 결과 정규분포 또는 정규분포와 비슷한 형태를 갖는 베타분포를 따른다. 그러므로 이와 같은 방식으로 생성된 데이터들은 호의 비용분포가 정규분포를 따른다고 가정한다.

<표 2> 일양분포 문제들의 MMO 통계치

n	max	\bar{X}	s
10	6	3.43	0.96
20	7	3.89	1.10
30	7	4.27	0.94
40	8	4.71	1.09
50	7	4.79	0.97
60	9	5.02	1.13
70	8	4.90	0.93
80	8	5.13	1.08
90	10	5.12	1.13
100	15	5.52	1.83
150	17	5.49	1.51
200	15	6.03	1.60
250	14	6.23	1.72
300	13	6.04	1.28

각 문제크기마다 50개씩 총 700개의 문제를 생성하여 MMO 값들을 구했다. 그리고 각 문제크기에 해당하는 50개의 MMO값들의 최대값, 표본평균값, 표본표준편차를 계산하였다. 실험을 통해 얻은 정규분포 문제들의 MMO 통계치는 <표 3>과 같다.

〈표 3〉 정규분포 문제들의 MMO 통계치

n	max	\bar{X}	s
10	8	4.58	1.28
20	12	5.72	1.65
30	10	6.54	1.50
40	11	7.22	1.49
50	12	7.20	1.96
60	18	7.54	2.57
70	13	7.58	2.23
80	15	7.60	1.91
90	17	8.28	2.49
100	20	9.12	2.61
150	18	10.08	2.31
200	27	10.48	3.56
250	24	11.72	3.34
300	25	12.88	4.22

4.2 회귀함수

4.1의 결과인 교점의 개수와 교점의 개수에 따른 MMO 값들의 $\bar{x} + 3s$ 를 이용하여 회귀함수를 구한다.

4.2.1 일양분포 문제의 회귀함수

[그림 4]는 호의 비용분포가 일양분포를 따르는 문제들의 MMO값들에 대한 전체분포와 회귀선을 대략적으로 보여준다. 회귀 분석을 통해 얻은 일양분포 문제의 회귀함수는 다음과 같다.

$$k = 6 + 0.33\sqrt{n} \quad (6)$$

회귀함수 식 (6)을 검증하기 전에 두 변수간의 상관계수를 구하고 5% 유의수준에서 두 변수간의 상관관계가 존재하는지를 알아보았다. 두 변수 즉, 교점의 개수와 MMO 통계치의 표본상관계수는 0.80853, p -value값은 0.0005이다. 식 (6)이 자료의 전체 산포 중에서 얼마나 설명하는가를 나타내는 R^2 는 0.7535로서, 전체 산포의 약 75%가 회귀직선에 의하여 설명된다.

식 (6)이 회귀모형의 기본가정을 잘 만족하는지를 판단하기 위해 오차항의 일반적인 가정들인 등분산성(constant variance), 정규성(normality), 독립성(independence)을 잔차(residual)를 이용하여 확인하였다. 오차항의 독립성을 검증하는 Durbin-Watson의 D 통계량 값이 1.802로 2에 가까운 값을 갖기 때문에 오차항은 독립적이라고 볼 수 있다. 오차항의 등분산성을 판단하는 잔차그림(residual plot)에서 잔차들이 랜덤하게 흩어져있고, 잔차들의 흩어진 폭이 예측값의 증감에 따라 변하지 않으며, 곡선형의 경향이 없기 때문에 식 (6)은 등분산성에 대한 가정을 만족한다. 정규성 검정을 수행하는 Shapiro-Wilk의 W 통계량과 p -value를 보면 식 (6)의 p -value가 0.1355로서 유의수준(0.05)보다 크기 때문에 잔차가 정규분포를 따른다고 할 수 있다. 따라서 회귀함수 식 (6)을 받아들여도 무방하다.

4.2.2 정규분포 문제의 회귀함수

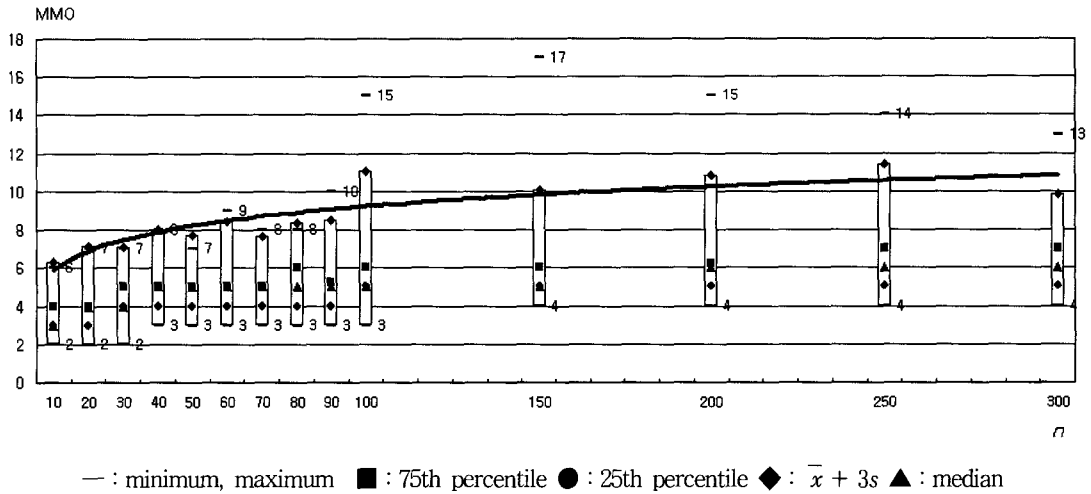
[그림 5]는 호의 비용 분포가 정규분포를 따르는 MMO값들에 대한 전체분포와 회귀선을 대략적으로 보여준다. 회귀분석을 통해 얻은 정규분포 데이터의 회귀함수는 다음과 같다.

$$k = 5.5 + 1.11\sqrt{n} \quad (7)$$

교점의 개수와 MMO 통계치의 표본상관계수는 0.96935, p -value값은 0.0001이며, 결정계수 R^2 는 0.9602으로서, 전체 산포의 약 96%가 회귀직선에 의하여 설명된다.

회귀함수 식(7)이 회귀모형의 기본가정을 만족하는 가를 알아보았다. Durbin-Watson의 D 통계량값이 2.464이다. 잔차그림(residual plot)에서 잔

차들이 랜덤하게 흩어져있고, 잔차들이 흩어진 폭이 예측값의 증감에 따라 변하지 않으며, 곡선형의 경향은 없다. 즉, 식 (7)은 등분산성에 대한 가정을 만족한다. Shapiro-Wilk의 W 통계량과 p -value를 보면 식 (7)의 p -value가 0.7335이다. 따라서 회귀분석을 통해 얻은 회귀함수 식 (7)을 받아들여도 무방하다.



[그림 4] 일양분포 문제들의 MMO값들에 대한 분포와 회귀선

5. 모형 검증

용을 비교하였다.

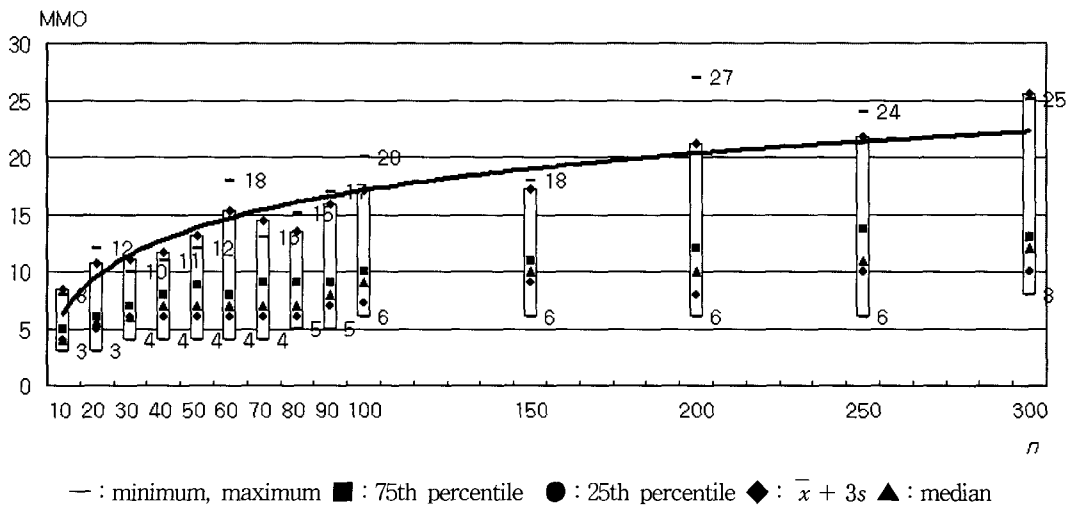
5.1 랜덤 생성 문제에 대한 회귀함수 검증

5.1.1 교점 개수가 100개인 문제

회귀함수의 적합성을 검증하기 위해 교점의 개수, 호 비용의 분포, 호 비용의 대칭성, 그리고 호 비용의 범위 등에 변화를 주어 문제를 생성하였다. 생성한 원문제의 최적 순환로의 비용과 모형을 적용한 호의 후보집합만으로 구한 최적 순환로의 비

호의 비용분포가 일양분포를 따르며, 다양한 호의 비용범위를 갖는 문제를 생성하였다. <표 4>에서 두 가지 비용범위를 갖는 문제들은 20개씩 생성하였고, 그 외의 문제들은 100개씩 생성하여 실험하였다.

<표 4>에서 동일성 여부는 원문제의 최적 순환



[그림 5] 정규분포 문제들의 MMO값들에 대한 분포와 회귀선

로의 비용과 식 (6)을 적용한 호의 후보집합만으로 구한 최적 순환로의 비용을 비교한 것으로 모두 동일하게 나타났다.

〈표 4〉 교점 개수가 100개인 랜덤 문제의 회귀함수 검증

분포	호 비용 범위	대칭성	동일성 여부
일양	[1~1000]	비대칭	동일함
일양	[1~20]	비대칭	
일양	[1~50]	비대칭	
일양	[1~50] or [1~100]	비대칭	
일양	[1~20] or [1~100]	비대칭	
일양	[1~20] or [1~1000]	비대칭	

5.1.2 교점 개수가 400개인 문제

호의 비용분포가 일양분포 또는 정규분포를 따르며, 호의 비용의 대칭성에 변화를 주어 문제를 생성하였다. <표 5>에서 대칭성이 대칭에 가까운 비대칭(quasi symmetric)의 문제들은 대칭인 문제들의 호 비용 값을 10% 내에서 변화를 주어 비대칭으로 생성하였다. 각 호의 비용 범위에 대해 10개씩 문제들을 생성하여 실험하였다.

〈표 5〉 교점 개수가 400개인 랜덤 문제의 회귀함수 검증

분포	호 비용범위	대칭성	동일성 여부
일양	[1 ~ 400]	비대칭	동일함
일양	[1 ~ 400]	대칭에 가까운 비대칭	
일양	[1 ~ 400]	대칭	
일양	[1 ~ 40]	대칭	
정규	[1 ~ 400√2]	비대칭	
정규	[1 ~ 400√2]	대칭	

<표 5>에서 일양분포 문제들은 식 (6)을 적용하여 후보집합을 구하였고, 정규분포 문제들은 식 (7)을 적용하여 호의 후보집합을 구하였다. 최적 순환로의 비용과 비교하여 호의 후보집합만으로 구한 최적 순환로의 비용은 모두 동일하게 나타났다.

5.1.3 다양한 호의 비용분포를 따르는 문제

호의 비용분포가 다양한 연속 분포의 형태를 따르는 비대칭 외관원문제들을 생성하였다. <표 6>에서 정규분포를 제외하고는 해당 분포를 따르도록 발생시킨 호 비용 값들이 대부분 한 자리 이하의 값이어서 범위 조정을 하였다. 각 분포의 모수값마다 교점의 개수가 400개인 문제를 10개씩 생성하여 실험하였다. 원문제의 최적 순환로의 비용과 식 (7)을 적용한 호의 후보집합으로부터 구한 최적 순환로의 비용을 비교한 결과 모두 동일하였다.

〈표 6〉 다양한 호 비용 분포를 따르는 비대칭 문제의 회귀함수 검증

분포	모수	호 비용 범위	동일성 여부
정규	$\mu = 200, \sigma = 5$	그대로 사용	동일함
	$\mu = 200, \sigma = 10$		
	$\mu = 200, \sigma = 50$		
지수	$\beta = 1$	발생값 × 100	동일함
	$\beta = 2$		
	$\beta = 3$		
	$\beta = 4$		
감마	$\alpha = 0.5, \beta = 1$	발생값 × 100	동일함
	$\alpha = 1, \beta = 1$		
	$\alpha = 2, \beta = 1$		
	$\alpha = 3, \beta = 1$		
	$\alpha = 4, \beta = 1$		
와이블	$\alpha = 0.5, \beta = 1$	발생값 × 100	동일함
	$\alpha = 1, \beta = 1$		
	$\alpha = 2, \beta = 1$		
	$\alpha = 3, \beta = 1$		
	$\alpha = 4, \beta = 1$		
로그노말	$\mu = 0, \sigma = 0.5$	발생값 × 100	동일함
	$\mu = 0, \sigma = 1$		
	$\mu = 0, \sigma = 1.5$		

또한, 호의 비용이 모수값의 변화에 따라 여러 가지 분포형태를 보여주는 베타분포를 따르는 비대칭 외관원문제들을 생성하였다. <표 7>에서 베타분포의 다양한 모수값에 따른 각 비용범위마다 교점의 개수가 400개인 문제를 10개씩 생성하였다.

원문제의 최적 순환로의 비용과 식 (7)을 적용한 호의 후보집합으로부터 구한 최적 순환로의 비용을 비교한 결과 모두 동일하게 나타났다.

랜덤하게 생성한 문제에 대한 모형검증을 통해 제안하는 호의 후보집합은 적어도 최적 순환로를 구성하는 모든 호를 포함한다. 또한 랜덤하게 생성한 문제가 일양분포인 경우 식 (6), 그 외의 문제들은 식 (7)을 적용하여 호의 후보집합을 구한다. 실험을 통해 이렇게 구한 호의 후보집합만을 이용해도 최적 순환로를 구할 수 있었다.

<표 7> 호 비용이 베타 분포를 따르는 비대칭 문제의 회귀함수 검증

분 포	모 수	호 비용의 범위	동일성 여부
베타	$\alpha_1=1.5, \alpha_2=5$	발생값 \times 400	동일함
	$\alpha_1=1.5, \alpha_2=3$	발생값 \times 400	
	$\alpha_1=5, \alpha_2=1.5$	발생값 \times 400	
	$\alpha_1=3, \alpha_2=1.5$	발생값 \times 400	
	$\alpha_1=5, \alpha_2=5$	발생값 \times 400	
	$\alpha_1=3, \alpha_2=3$	발생값 \times 400	
	$\alpha_1=2, \alpha_2=2$	발생값 \times 400	
	$\alpha_1=1, \alpha_2=1$	발생값 \times 400	
	$\alpha_1=0.8, \alpha_2=2$	발생값 \times 400	
	$\alpha_1=1, \alpha_2=2$	발생값 \times 400	
	$\alpha_1=2, \alpha_2=1$	발생값 \times 400	
	$\alpha_1=2, \alpha_2=0.8$	발생값 \times 400	
	$\alpha_1=0.8, \alpha_2=0.2$	발생값 \times 400	
	$\alpha_1=0.5, \alpha_2=0.5$	발생값 \times 400	

5.2 TSPLIB 문제에 대한 회귀함수 검증

현실문제인 TSPLIB 문제를 사용하여 회귀함수를 검증하였다. 각 문제의 최적 순환로를 구성하는 호들의 MMO값과 식 (7)을 적용한 k를 비교하였으며 결과는 <표 8>과 같다.

<표 8> TSPLIB 문제의 회귀함수 검증

problem	n	MMO	k	비율 (%)
br17	17	3	12	100.0
ft53	53	10	14	36.5
ft70	70	11	15	30.6
ftv33	34	6	12	46.0
ftv35	36	5	12	43.5
ftv38	39	4	13	43.0
ftv44	45	4	13	36.0
ftv47	48	6	13	33.9
ftv55	56	10	14	31.6
ftv64	64	6	15	29.5
ftv70	71	7	15	26.9
ftv90	91	7	16	21.4
ftv100	101	7	17	20.4
ftv110	111	7	17	18.6
ftv120	121	7	18	18.0
ftv130	131	14	18	16.5
ftv140	141	17	19	16.1
ftv150	151	17	19	15.0
ftv160	161	17	20	14.9
ftv170	171	7	20	14.0
krol124p	100	7	17	19.9
p43	43	8	13	44.6
rbg323	323	12	25	100.0
rbg358	358	22	27	100.0
rbg403	403	21	28	100.0
rbg443	443	21	29	100.0
ry48p	48	9	13	33.2

<표 8>에서 비율은 원문제에서의 호의 개수에 대비 k로부터 구성된 호의 후보집합의 호의 개수의 비율이다. 27개의 문제들 모두 식 (7)로부터 최적 순환로를 보장하는 호의 후보집합을 구성할 수 있다. br17은 동일한 순위를 갖는 호들이 많아 전체 호를 호의 후보집합에 포함해야 한다. 스택커 크레인 문제인 rbg323, rbg358, rbg403, rbg443도 동일 순위의 호들이 많아 전체 호를 호의 후보집합에 포함하여야 한다. 그러나 문제의 특성상 이 4개

의 문제들은 MMO값까지의 호들로 호의 후보집합을 구성하면 모든 호를 포함하는 것과 동일하기 때문에 제안하는 후보집합의 적용여부와 관계없이 최적 순환로를 구하기 위해서는 모든 호를 호의 후보집합에 포함해야 한다.

TSPLIB 문제의 경우 제안하는 호의 후보집합을 사용할 경우 평균 41.1%(스택커 크레인문제를 제외할 경우 30.9%)의 호들만으로 최적 순환로를 구할 수 있다.

6. 결 론

본 연구에서는 비대칭 외판원문제의 호의 후보집합을 결정하는 회귀함수를 제시하였으며, 이 호의 후보집합만을 사용함으로써 모든 호를 고려하지 않고 최적 순환로를 구할 수 있음을 실험을 통해 보였다. 호의 비용이 대칭인 외판원문제에 한정되어 연구되어진 기존의 후보집합과는 달리, 본 연구에서는 비대칭 외판원문제에 초점을 맞추었으며, 다양한 실험을 통해 제안하는 회귀함수가 비대칭 외판원문제 뿐만 아니라 대칭 외판원문제에도 적용 가능함을 보여주었다. 또한 TSPLIB의 문제들에 대한 모형검증을 통해 현실문제에도 적용 가능함을 보였다.

호의 분포가 일양분포를 따를 경우 식 (6)을 적용하며, 일양분포 이외의 분포 또는 호의 분포를 알 수 없을 경우 식 (7)을 적용하여 구성된 호의 후보집합으로부터 최적 순환로를 구할 수 있다. 향후에는 제시한 호의 후보집합을 적용하여 계산시간을 단축하는 연구가 필요할 것으로 판단된다.

참 고 문 헌

- [1] 강맹규, 「네트워크와 알고리즘」, 박영사, 1991.
- [2] Bentley, J.L., "K-d Trees for Semidynamic Point Sets," *Sixth Annual ACM Symposium on Computational Geometry*, Berkeley, (1990), pp.187-197.
- [3] Cirasella, J., D.S. Johnson, L.A. McGeoch, and W. Zhang, "The Asymmetric Traveling Salesman Problem : Algorithms, Instance Generators, and Tests," *Springer Lecture Notes in Computer Science*, 2153, Springer-Verlag, Berlin, (2001), pp.32-59.
- [4] Hidaka, K., "Simulation-Based Approach to the Warehouse Location Problem for a Large-Scale Real Instance," *1997 Winter Simulation Conference*, (1997), pp.1214-1221.
- [5] Johnson, D.S., "Local Optimization and the Traveling Salesman Problem," *Proc. 17th Colloquium on Automata, Languages and Programming*, edited by G. Goos and J. Harmanis, Springer-Verlag, Heidelberg, (1990), pp.446-461.
- [6] Junger, M., G. Reinelt, and G. Rinaldi, "The Traveling Salesman Problem," *Network Models*, edited by M.O. Ball, T.L. Magnanti, C.L. Monma, and G.L. Nemhauser, *Handbooks in Operations Research and Management Science*, Elsevier.
- [7] Laporte, G., "The Traveling Salesman Problem : An Overview of Exact and Approximate Algorithms," *European Journal of Operational Research*, Vol.59(1992), pp. 1234-1247.
- [8] Reinelt, G., *The Traveling Salesman Computational Solutions for TSP Applications*, Springer-Verlag, Berlin, 1994.
- [9] Reinelt, G., "TSPLIB-A Traveling Salesman Problem Library," *ORSA Journal of Computing*, Vol.3(1991), pp.376-384.
- [10] Sakakibara, K., "New Edges Not Used in Shortest Tours of TSP," *European Journal of Operational Research*, Vol.104(1998), pp.129-138.