

# A Heuristic for Dual Mode Routing with Vehicle and Drone

Yun-Hong Min \*, Yerim Chung \*\*

## Abstract

In this paper we consider the problem of finding the triplet  $(S, \pi, f)$ , where  $S \subseteq V$ ,  $\pi$  is a sequence of nodes in  $S$  and  $f: V \setminus S \rightarrow S$  for a given complete graph  $G = (V, E)$ . In particular, there exist two costs,  $c_{uv}^V$  and  $c_{uv}^D$  for  $(u, v) \in E$ , and the cost of triplet  $(S, \pi, f)$  is defined as  $\sum_{i=1}^{|\pi|} c_{\pi(i)\pi(i+1)}^V + 2 \sum_{u \in V \setminus S} c_{uf(u)}^D$ . This problem is motivated by the integrated routing of the vehicle and drone for urban delivery services. Since a well-known NP-complete TSP (Traveling Salesman Problem) is a special case of our problem, we cannot expect to have any polynomial-time algorithm unless P=NP. Furthermore, for practical purposes, we may not rely on time-exhaustive enumeration method such as branch-and-bound and branch-and-cut. This paper suggests the simple heuristic which is motivated by the MST (minimum spanning tree)-based approximation algorithm and neighborhood search heuristic for TSP.

▶ Keyword : Vehicle and Drone, Heuristic, Traveling Salesman Problem, Minimum Spanning Tree, Neighborhood Search

## 1. Introduction

지금까지 도심 지역에서의 배송 서비스는 주로 차량에만 의존해 왔다. 하지만, 도심 지역의 복잡한 도로 네트워크와 교통 혼잡은 차량을 이용한 물류 서비스의 품질을 저해하는 주요 원인으로 지적되어 왔다 [1]. 전통적으로 이륜차 서비스가 이를 부분적으로 해결하는 대안으로 제시되어 활용되고 있지만, 높은 서비스 비용은 소비자들에게도 부담으로 작용하고 있어서, 새로운 대안에 대한 관심이 높아지고 있다.

복잡한 도로 네트워크와 교통 혼잡을 피할 수 있다는 점에서 드론을 이용한 배송 서비스가 최근 큰 주목을 받고 있다 [2-4]. 하지만, 드론은 상대적으로 작은 배송 용량과 빈번히 요구되는 충전으로 인해 기존의 차량 중심의 배송 서비스를 완전히 대체하기는 어려울 것으로 보인다. 따라서 기존에 이용되어 왔던 차량과 새롭게 등장한 드론을 혼용하는 방법은 높은 도입 가능성을 갖는다고 볼 수 있다.

본 논문에서는 차량과 드론을 혼용하는 배송 서비스의 효율적인 운용을 다음과 같은 그래프 위에서의 조합 최적화 (combinatorial optimization) 문제로 모델링 한다. 배송 목적지를 노드로 갖는 완전 연결 그래프 (complete graph)를  $G = (V, E)$ 라고 하자. 임의의 두 노드  $u$ 와  $v$  사이에는 두 종류의 비용이 존재한다. 하나는 두 노드 사이를 차량으로 이동할 때 발생하는 비용  $c_{uv}^V$ 이고 다른 하나는 두 노드 사이를 드론으로 이동하는 비용  $c_{uv}^D$ 이다.  $V$ 의 부분 집합  $S$ ,  $S$ 에 속한 노드들로 구성된 회로  $\pi$ ,  $V \setminus S$ 에서  $S$ 로의 함수  $f: V \setminus S \rightarrow S$ 가 주어지면, 다음과 같이 비용이 결정된다. 먼저, 회로  $\pi$ 는 이를 구성하는 호(edge)들의  $c_{uv}^V$  값의 합만큼의 비용을 발생시키고,  $f$ 는  $c_{uf(u)}^D$  값의 합의 두 배의 비용을 발생시킨다. 따라서 전체 비용은 식 (1)과 같다:

• First Author: Yun-Hong Min, Corresponding Author: Yerim Chung  
\*Yun-Hong Min (yunhong.min@gmail.com), Samsung Advanced Institute of Technology, Samsung Electronics.  
\*\*Yerim Chung (yerimchung@yonsei.ac.kr), School of Business, Yonsei University  
• Received: 2016. 08. 26, Revised: 2016. 09. 02, Accepted: 2016. 09. 07.

$$\sum_{i=1}^{|S|} c_{\pi(i)\pi(i+1)}^V + 2 \sum_{u \in V \setminus S} c_{uf(u)}^D \quad (1)$$

식 (1)의 첫 번째 항은 차량을 이용한 이동에 의해 발생하는 비용의 합을 의미하고, 두 번째 항은 드론의 이동에 의해 발생하는 비용의 합을 의미한다. 이 때 드론의 경우에는 제한된 용량과 사고의 위험으로 인해, 특정 배송지로 배송을 완료한 경우, 다시 출발지 (차량이 정차하고 있는 거점)로 되돌아 온 다음, 다음 배송지로 이동하는 상황을 가정한다.

즉, 이 조합 최적화 문제는 차량으로 주요 거점에 배송을 수행하고, 특정 배송지로부터 인접한 몇 개의 배송지에는 드론을 통해서 배송을 수행하는 상황을 묘사한다.

이 모형에서 각 호에 정의되어 있는 드론 이동 비용  $c_{uv}^D$ 를 매우 크게 설정하게 되면, 잘 알려진 NP-complete 문제인 TSP(Traveling Salesman Problem)와 동일한 문제가 된다. 따라서, 우리 모형은 P=NP가 아니라는 가정에서 다항시간 알고리즘(polynomial-time algorithm)을 기대할 수 없다 [5].

우리 문제를 포함한 많은 NP-complete 또는 NP-hard 문제는 정수 계획법(Integer Programming)으로 모형화 되며, Branch-and-Bound와 같은 효율적인 나열법(enumeration method)를 활용하여 최적 해를 구할 수 있다 [6]. 하지만, 이러한 접근 방법은 많은 계산 시간을 요구하기 때문에 빠른 계산을 요구하는 현실 문제에서는 어느 정도의 해 품질의 저하를 수용하면서 대신 더 빠르게 해를 찾는 대안을 선택하기도 한다 [7]. 뿐만 아니라, 좋은 성능의 가능 해는 최적 해를 찾기 위한 초기해로도 활용 가능하다.

본 논문은 이러한 관점에서 해를 얻기까지 많은 시간을 요구하는 나열법을 피하고 빠른 시간에 해를 찾을 수 있는 휴리스틱을 제안한다. 2장에서는 차량과 드론을 혼용하는 기존 방법들을 소개한다. 3장에서는 휴리스틱을 제안하고 4장에서는 제안된 휴리스틱을 무작위로 생성된 데이터에 적용하여 성능을 평가해 본다. 5장에서는 이 논문에 대한 향후 연구 방향을 소개한다.

## II. Related Works

차량과 드론을 동시에 운용하는 배송 서비스에 대한 연구는 크게 차량이 드론의 이동기지 역할을 하는 FSTSP (Flying Sidekick TSP) 접근 방법 [8]과 차량과 드론이 독립적으로 운영 되는 TSP-D(Traveling Salesperson Problem with Drone) [9]가 있다. 본 논문은 차량이 드론의 이동기지 역할을 하는 상황을 함수  $f$ 로 묘사했다는 점에서 FSTSP와 같은 상황을 다룬다. 하지만, FSTSP는 차량에서 출발한 드론이 여러 배송지를 방문한 다음, 차량으로 돌아오는 상황을 다룬다. 하지만 차량과 드론이 이와 같은 방법으로 운용되기 위해서는 차량과 드론을 동시에 운용할

필요가 있기 때문에, 두 명 이상의 작업자가 함께 이동을 해야 되는데 이러한 방법은 높은 인건비가 필요하다. 뿐만 아니라, 드론의 제한적인 용량과 사고 시 발생할 수 있는 높은 비용 등을 고려했을 때, 드론이 여러 배송지를 거치는 것보다 한 배송지의 방문한 이후에 차량으로 돌아오는 시나리오가 더 안정적인 운용 전략이 될 수 있다. 따라서, [8]보다 간단하게 보이는 본 논문의 운용 시나리오도 현실적으로 의미가 있다고 할 수 있다.

배송 네트워크에서 드론을 운용하는 연구는 2015년 이후로 위에서 언급된 소수의 문헌들만 존재한다. 하지만 현실에서는 시험적으로 드론을 이용한 배송 서비스가 제공되고 있다. 대표적인 예는 아마존의 ‘프라임 에어(Prime Air)’라는 서비스로 드론을 이용해서 미국 도심 일부 지역에서 30분내 배송을 목표로 제공하고 있다 [10].

## III. The Proposed Heuristic

1장에서 언급한 바와 같이 우리 문제는 TSP를 포함한다. 중요한 관찰 중 하나는 집합  $S$ , 즉 해밀토니안 회로(Hamiltonian Circuit)을 구해야 할 노드들의 집합이 결정되면 나머지 노드들을 담당할 드론의 출발지, 즉  $S$ 에 속한 노드들 중,  $S$ 에 속하지 않는 임의의 노드를 담당할 노드는 쉽게 결정된다는 것이다. 따라서 이 문제를 풀기 위한 자연스러운 접근 방법 중 하나는 해밀토니안 회로를 찾을 집합  $S$ 를 먼저 구한 다음, 나머지 노드들을  $S$ 에 속한 노드들로 할당하는 2단계로 구성된 방법을 사용하는 것이다. 하지만,  $S$ 가 미리 정해져 있어도 TSP의 최적해를 구하는 것은 어렵다 [5]. 따라서, 본 논문은 TSP에 대한 근사해법 (approximation algorithm) 중 간단하면서도 좋은 해 품질을 보장하는 최소비용 나무 (minimum spanning tree, 최소걸침나무)를 이용한 근사해법을 휴리스틱의 서브루틴으로 사용 할 것이다. 이 근사해법은 항상 최적 해의 2배를 넘지 않는 목적함수를 갖는 가능 해를 구하는 것을 이론적으로 보장 한다 [7].

또 하나의 아이디어는 TSP를 위한 휴리스틱에서 흔히 사용되는 이웃 탐색(neighborhood search) 기법을 활용하는 것이다 [11]. 이웃 탐색 기법은 특정 해의 미리 정해진 이웃 중에서 더 좋은 해를 찾는 방법이기 때문에 branch-and-bound 기법과 같은 나열법에 대한 하나의 대안이 될 수 있다 [12]. 본 논문에서는 현재 해  $(S, \pi, f)$ 에 대한 이웃  $(S', \pi', f')$ 는  $v \in S$ 에 대해 다음과 같이 정의 한다:  $\pi(k) = v$ , 즉 현재 해에서 해밀토니안 회로의  $k$ 번째 노드가  $v$ 라고 하자.

- $S' = S \setminus \{v\}$ ,
- $\pi'(i) = \pi(i), i = 1, \dots, k-1$
- $\pi'(i) = \pi(i+1), i = k, \dots, |S|-1$
- $f'(u) = f(u), u \in V \setminus S$

•  $f'(v) = u^*, u^* = \operatorname{argmin}_{u \in S} c_{vu}^D$

즉, 해밀토니안 순회에 포함된 임의의 노드  $v$ 에 의해 정의되는 이웃은 1) 순회에서  $v$ 를 제거하여 새로운 순회를 만들고, 2)  $v$ 를 남아 있는 해밀토니안 순회에 속한 다른 노드 중에서 가장 작은 드론 이동 비용으로 연결이 가능한 노드에 할당하는 방법으로 정의한다.

현재 해를 위에서 정의한 새로운 해로 대체할 때 비용의 변화는 식(2)와 같이 정의된다.

$$\Delta = 2c_{vu}^D + c_{\pi(k-1)\pi(k+1)}^V - c_{\pi(k-1)\pi(k)}^V - c_{\pi(k)\pi(k+1)}^V$$

위에서 언급한 두 관찰을 기반으로 우리는 다음과 같은 2단계의 휴리스틱을 제안한다 (Fig. 1 참고).

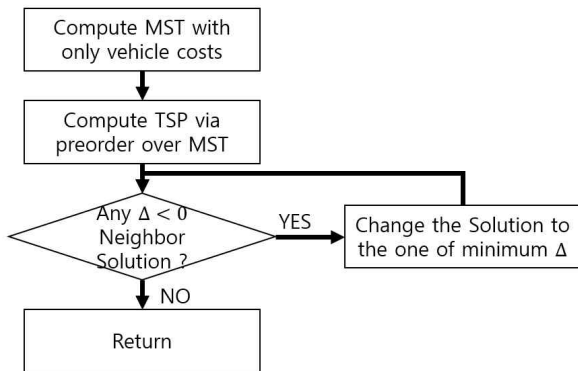
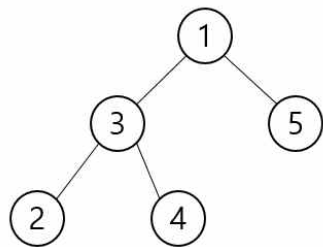


Fig. 1. Flow chart for the proposed heuristic

Step 1. 각 호  $(u, v) \in E$ 의 비용을  $c_{uv}^V$ 로 하는 최소비용 나무를 계산한 다음 [13], 노드들을 최소비용나무에서의 전위 순회(preorder) 순서로 나열하여 해밀토니안 회로를 구한다 (Fig. 2 참고).

Step 2. 현재 해의 이웃 중에서  $\Delta$  값이 음수이면서 가장 작은 이웃을 선택하여 이를 새로운 해로 정한다. 만약 모든 이웃의  $\Delta$  값이 0 이상인 경우에는 종료하고 그렇지 않은 경우에는 Step 2를 반복한다. 이웃을 계산할 때, 이미 드론 노드와 연결이 되어 있는 노드의 경우에는 이웃 계산에서 제외한다.



preorder: (1,3,2,4,5)

Fig. 2. An example of preorder traversing

위의 휴리스틱은 종료 이전에 항상 우리 문제의 가능해(feasible solution)를 유지한다는 특징이 있다. 또한 Step 1에서 최소비용나무를 계산하는 과정은  $O(|E|)$ 가 필요하다. Step 2에서는 매번  $S$ 의 크기가 적어도 하나는 감소하므로 최악의 경우  $|V|$ 번의 반복횟수가 필요하며 이웃탐색 횟수는  $|V|$ 을 넘지 않으므로 역시  $O(|E|)$ 의 복잡도를 갖는다 (왜냐하면 완전 연결 그래프(complete graph)를 가정하기 때문).

무작위로 생성된 6개의 노드로 이루어진 Table 1의 예를 살펴보자. 이 예는 6개의 노드로 구성된 그래프의 인접행렬 표현이다.  $(i, j)$ -번째 셀의 순서쌍은 순서대로 노드  $i$ 에서 노드  $j$ 로 차량이 이동할 때 발생하는 비용과 드론이 이동할 때 발생하는 비용을 의미한다. 예를 들어, 두 번째 행의 세 번째 열의 값 이동할 때 발생하는 (4,5)는 두 번째 노드에서 세 번째 노드로 이동할 때, 차량은 4의 비용을 발생시키고, 드론은 5의 비용을 발생시킨다는 것을 의미한다.

Table 1. A toy example

	(5,2)	(6,3)	(6,-)	(7,4)	(4,-)
(5,2)		(4,-)	(6,2)	(7,2)	(8,2)
(6,3)	(4,-)		(9,-)	(5,2)	(7,4)
(6,-)	(6,2)	(9,-)		(8,1)	(9,2)
(7,4)	(7,2)	(5,2)	(8,1)		(6,-)
(4,-)	(8,2)	(7,4)	(9,2)	(6,-)	

Table 1의 예에서는 드론의 이동 비용이 '-'로 표기되어 있는 경우가 있다. 이 경우는 드론으로는 이동이 불가능한 상황 등을 묘사한 것으로, 매우 큰 수를 비용으로 갖는다고 가정할 수 있다.

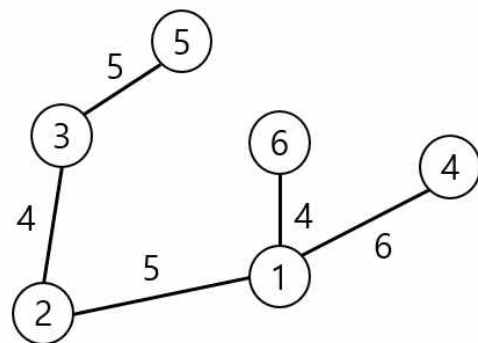


Fig. 3 Minimum Spanning Tree for Table 1.

제안한 휴리스틱의 Step 1에서는 차량 이동 비용만을 이용한 최소결침나무를 계산한 다음, 노드들의 전위순회를 계산한다. Fig 2는 예제 문제의 최소비용나무이며, 이 때 전위순회는 (5,3,2,1,6,4)이고 이에 해당하는 해밀토니안 회로는 (5,3,2,1,6,4,5)가 되고 비용은  $5+4+5+4+9+8=35$ 이다.

Step 2에서는 현재 해의 이웃들로 해를 치환할 때 발생하는 비용 변화를 계산한다. 현재 해에서 이웃별 비용 변화는 Table 2의 두 번째 열에 요약하였다.

Table 2의 두 번째 열에 의하면, 노드 4에 의해 정의되는 이웃이 가장 비용 개선이 크다. 따라서 이 이웃을 다음 해로 설정한다. 즉, (5,3,2,1,6,5)가 해밀토니안 회로와 노드 4를 노드 5로 연결하는 것이 새로운 해가 된다. 다음 단계로 넘어가기 위해서는 이 해로부터 다시 이웃과 이웃별 비용 변화를 계산해야 한다. 그 결과는 Table 2의 세 번째 열과 같다.

Table 2. Iterations for the toy example in Table 1.

	iteration 1	iteration 2
1	$-5-4+8+2+2=3$	$-5-4+8+2+2=3$
2	$-4-5+6+2+2=1$	$-4-5+6+2+2=1$
3	$-5-4+7+2+2=2$	$-5-4+7+2+2=2$
4	$-9-8+6+1+1=-9$	-
5	$-5-8+9+1+1=-3$	$-5-6+7+1+1=-2$
6	$-4-9+6+2+2=-3$	$-4-6+7+2+2=1$

Table 2의 세 번째 열에 의하면, 새로운 해는 해밀토니안 회로 (3,2,1,6,3)으로 정의된다. 그리고 해로부터 다시 이웃별 비용 개선을 계산한 결과는 Table 2의 세 번째 열과 같다. 하지만, 이전 스텝에서 노드 4가 드론 노드로 할당이 되면서 노드 5와 연결되었으므로 노드 5에 의해 정의되는 이웃을 새로운 해로 치환할 경우 비용 개선이 있다고 하더라도 이 노드는 이웃 계산에서 제외되어야 한다. 즉, 알고리즘은 종료된다. 이 때, 비용은 해밀토니안 회로에 의해 정의되는 비용  $5+4+5+4+6=24$ 와 드론에 의해 발생하는 비용  $1+1=2$ 를 합한 26의 비용이 발생한다.

제안한 휴리스틱의 성능 평가를 위해서는 [8]과 같이 문제의 혼합정수계획 모형을 만든 다음, 상용 소프트웨어를 이용하여 최적 해를 구하여 성능을 비교할 수 있다. 하지만 우리는 다음의 관찰을 사용하여 최적 해의 하한을 구해서 휴리스틱의 성능을 비교할 것이다. 우리 문제의 최적 해에서 차량 이동 호 몇 개를 제거하여 나무(tree)를 만들 수 있다. 따라서, 각 호의 비용을 차량 이동 비용과 드론 이동 비용 중 작은 값으로 정하여 최소비용나무를 구하면 우리 문제의 최적 해의 최적 비용의 하한을 구할 수 있다. 다음 장에서는 이 하한과 제안한 휴리스틱의 성능을 비교할 것이다.

### IV. Evaluations

제안된 휴리스틱의 성능을 평가하기 위해 무작위로 생성한 10개의 예제에 대하여 차량만을 이용할 때의 비용과 차량과 드론을 혼용하여 운용했을 때의 비용을 비교하였다.

[8]은 20개 수준의 노드에 대한 실험 결과를 제시하였다. 그 이유는 20개 수준의 노드로 정의되는 혼합정수계획 모형이 매우 크기 때문에 상용 소프트웨어로 최적 해를 구하기가 어렵기 때문이었다. 이 논문에서 제안한 휴리스틱은 혼합정수계획 모형에 의존하는 상용 소프트웨어를 사용하지 않기 때문에 이보다 더 많은 노드 개수를 갖는 무작위로 생성된 예들에 대해 실험하였다.

각 예제는 한 번의 길이가 50인 정사각형 안에 무작위로 생성된 100개의 노드로 구성되어 있다. 차량으로 이동할 때 발생하는 비용은 유클리디안(Euclidean) 거리로 정의하였다. 드론은 유클리디안 거리로 4 이내인 노드 사이만을 이동할 수 있다고 가정하였고, 드론의 이동 비용은 차량 이동 비용에 0.01과 0.2 사이에 무작위로 선택된 비용을 곱한 만큼의 비용이 발생한다고 가정하였다. 즉, 두 노드 사이에 드론 운용이 가능한 경우 드론 이동 비용은 항상 차량 이동 비용의 0.2배 보다는 같거나 작고, 0.01배보다는 같거나 크다.

Table 3에서 10개의 예제에 대해 식(1)의 비용을 차량만 운용한 경우와 차량과 드론을 함께 운용한 경우의 결과를 비교하였다. 차량만 운용할 때의 비용은 최소비용나무를 이용한 TSP의 2-근사해법으로 구한 비용이다. 이 비용은 TSP의 최적해의 2배를 넘지 않는다. Table 3의 세 번째 열은 상대적인 에러 감소 비율을 의미하며 (첫번째 열-두번째 열)/첫번째 열을 백분율로 표기한 것이다.

Table 3. Simulation Result

	2-OPT TSP	Heuristic	Relative Error Reduction
#1	509.17	404.09	20.63
#2	509.97	411.29	19.35
#3	521.62	440.75	15.50
#4	519.57	406.86	21.69
#5	498.75	400.84	19.63
#6	548.46	468.27	14.62
#7	601.79	483.59	10.64
#8	522.88	417.91	20.07
#9	516.92	436.96	15.46
#10	571.05	493.55	13.57

Table 4는 10개의 예제에 대해 최적 해의 목적 함수 값의 하한과 휴리스틱의 결과를 비교한 것이다.

Table 4. Comparisons with lower bound

	Lower Bound	Heuristic	Approximation Ratio
#1	212.18	404.09	1.90
#2	215.91	411.29	1.90
#3	223.63	440.75	1.97
#4	191.99	406.86	2.11
#5	227.7	400.84	1.76
#6	236.14	468.27	1.98
#7	245.92	483.59	1.96
#8	193.79	417.91	2.15
#9	227.34	436.96	1.92
#10	241.25	493.55	2.04

Table 4의 세 번째 열은 휴리스틱의 비용을 하한 값으로 나타낸 결과이다. 최소비용나무를 이용한 TSP의 휴리스틱이 최적 해의 2배를 넘지 않음을 보장하는 근사 해법이지만, 이를 이용한 우리 휴리스틱은 근사 계수는 2를 넘는 경우도 있다 (4, 8번째 예제). TSP에 대한 근사해법으로 구한 해의 성능은 모두 하한의 2배를 넘었다. 이는 하한이 지나치게 보수적이기 때문일 수도 있고, 우리 문제는 드론을 사용하면서 일반적인 TSP 문제보다 더 작은 비용으로 노드들을 방문할 수 있기 때문일 수도 있다. 하지만, 우리 휴리스틱은 10개 예제 중 8개의 예제에서 하한의 2배를 넘지 않는 비용을 발생시키는 해를 구할 수 있었다.

## V. Conclusions

본 논문에서는 차량과 드론을 혼용하는 배송 서비스를 모형화하는 조합 최적화 문제를 제안하고, 문제의 계산적인 어려움과 현실 문제에서 요구되는 컴퓨팅 타임을 고려한 새로운 휴리스틱을 제안하였다.

이 휴리스틱은 잘 알려진 TSP의 근사해법인 최소비용나무를 이용한 근사해법으로 얻은 해밀토니안 순회를 초기해로 한다. 그리고, 문제의 특수한 구조를 기반으로 이웃해를 정의한 다음, 가장 좋은 이웃 해를 찾아서 해를 개선시켜 나가는 지역 탐색 기법을 사용한다.

실험 결과, 초기해로부터 의미 있는 개선을 얻을 수 있었고, 최적 목적 값의 하한과의 비교에서도 하한과 2배에 근접한 근사 계수를 얻을 수 있었다. 이 오차가 보수적인 하한을 사용한 결과인지, 아니면 이론적으로 더 큰 오차를 보일 수도 있는지 규명하는 것은 미결 문제(open question)이다.

이 문제와, 모형, 그리고 제안된 휴리스틱과 관련된 향후 연구 방향으로는 다음과 같은 것들이 가능하다.

모형 관점에서 제안된 모형이 표현하지 못하는 현실적인 상

황을 파악하고, 이를 반영하여 모형을 확장하는 연구가 필요하다. 또한 확장된 모형에 맞는 새로운 해법의 개발이 필요할 수도 있다.

본 논문이 제안한 문제의 혼합정수계획 모형을 개발하고, 이를 이용하여 최적 해를 구하여 휴리스틱의 성능을 엄밀하게 평가하는 것이 필요하다. 뿐만 아니라, 혼합정수계획 모형을 기반으로 효율적인 절단면(cut)을 이용하여 최적 해를 구하는 성능을 향상시키는 연구도 의미 있다고 할 수 있다.

## REFERENCES

- [1] M. Savelsbergh and T. van Woensel, "City logistics: Challenges and opportunities". *Transportation Science*, Vol. 50, No. 2, pp. 579-590, May. 2016.
- [2] Amazon and drones - here is why it will work, <http://www.forbes.com/sites/stevebanker/2013/12/19/amazon-drone-here-is-why-it-will-work/>
- [3] Drone delivery: DHL 'parcelcopter' files to German isle, <http://www.reuters.com/article/2014/09/24/us-deutsche-post-drones-idUSKCN0HJ1ED20140924>
- [4] Like Amazon, UPS Also Considering Using Unmanned Flying Vehicles, <http://abcnews.go.com/Technology/amazon-ups-drone-delivery-options/story?id=21086160>
- [5] M. R. Garey and D. S. Johnson. "Computers and Intractability; A guide to the Theory of NP-Completeness" W. H. Freeman & Co., 1990
- [6] G. Nemhauser and L. A. Wolsey, "Integer and Combinatorial Optimization" John Wiley & Sons, 1988
- [7] V. V. Vazirani. "Approximation Algorithms" Springer-Verlag Berlin Heidelberg. 2003
- [8] C. C. Murray and A. G. Chu, "The flying sidekick traveling salesman problem: Optimization of drone-assisted parcel delivery", *Transportation Research Part C: Emerging Technologies*, Vol. 54, pp. 86-109, May. 2015.
- [9] N. Agatz, P. Bouman, M. Schmidt, "Optimization approaches for the traveling salesman problem with drone", *ERIM Report Series Reference*, No. ERS-2015-011-LIS, 2015.
- [10] Amazon Prime Air Is Here, but Not in the Expected Form, <http://fortune.com/2016/08/05/amazon-prime-air-car-go/>

- [11] G. A. Croes, "A method for solving traveling salesman problem", Operations Research. Vol. 6, No. 6, pp. 791-812, Nov. 1958.
- [12] Jun-Ha Hwang, "An integer programming-based local search for the set covering problem", Journal of The Korea Society of Computer and Information. Vol. 19, No. 10, Oct. 2014.
- [13] Ju-Young Lee, "An efficient implementation of Kruskal's algorithm for a minimum spanning tree", Journal of the Korea Society of Computer and Information. Vol. 19, No. 7, Jul. 2014.

### Authors



Yun-Hong Min received the B.S. degree in Industrial & Management Engineering from POSTECH, and Ph.D. degree in Industrial Engineering from Seoul University in 2006 and 2012, respectively.

Dr. Min joined the research staff member of Samsung Advanced Institute of Technology (SAIT) in 2012. He is currently a research staff member in SAIT. He is interested in mathematical programming and machine learning.



Yerim Chung received the B.S. degree in Business Administration from Yonsei University, Korea, in 2000. She received the M.S. and Ph.D. degree in Applied Mathematics and Computer Science from Paris 1 University, France, in 2004 and

2010, respectively. Dr. Chung joined the faculty of Business School at Yonsei University, Seoul, Korea, in 2011. She is interested in Inverse Optimization, Transportation Science and Network Effects.