

# 차고지의 유연한 할당을 고려한 택배 수거 경로 계획

김현준 · 김예찬 · 김병수<sup>†</sup>

인천대학교 산업경영공학과

## Pick-up Service Vehicle Routing Problem With Flexible Assignment of Depots

Hyeon Jun Kim · Ye chan Kim · Byung Soo Kim

Department of Industrial Management Engineering, Incheon National University

We address a vehicle routing problem for pick-up services, with flexible assignment of decentralized depots by utilizing parking lots throughout urban areas. The problem involves determining vehicle routes, which include deciding which demand locations each vehicle will visit, as well as depot assignments, while minimizing the operational costs including travel cost, parking cost, and tardiness cost. A mixed integer linear programming model is formulated to optimally solve the small-sized problem instances. To solve the problem efficiently, we propose a three-stage heuristic including clustering-based approach to group demand locations. For large-sized problem instances, a three-stage solution approach is proposed: (1) clustering demand locations into small-sized groups, (2) assignment of clusters to vehicles based on the distance between each vehicle and the cluster centroid, and (3) determining optimized routes for each vehicle by applying a meta-heuristic algorithm with iterative cluster reassignment. The effectiveness of the proposed approach is demonstrated through computational experiments showing significant cost reductions compared to conventional single-depot operations. This study provides operational insights into sustainable urban logistics and highlights the benefits of flexible depot utilization to enhance operational efficiency.

**Keywords:** Urban Logistics, Pick-up Service Vehicle Routing, Mixed Integer Linear Programming, Clustering Algorithm, Meta-heuristic

### 1. 서론

인천광역시의 소상공인 중 다수는 현재 월평균 10개 내외의 소량 택배 발송 물량을 보유하고 있어 개별 계약 시 높은 운송 단가가 책정되는 문제가 발생한다. 이를 해결하기 위해 인천광역시는 소상공인들의 분산된 물량을 취합하여 일괄적으로 택배사와 계약하는 ‘소상공인 택배 지원사업’을 운영하고 있다. 해당 사업은 현재 소상공인을 대상으로 시행되고 있으나, 2027년까지 인천 시민 전체로 대상을 확대하여 연간 약

18,000,000건의 물량을 처리하는 것을 목표로 하고 있다 (Incheon Metropolitan City, 2024a). 현재 이 사업은 인천광역시가 택배 수거를 담당하고, 계약된 민간 택배회사가 택배 배송을 담당하는 구조로 운영된다. 인천광역시가 자체적으로 운영하는 전기 화물차를 통해 고객을 방문하여 택배를 수거하고, 수거된 물량을 집화센터에 집화하면, 이후 계약된 민간 택배회사가 집화 물량을 수거하여 최종 배송을 담당한다(Incheon Metropolitan City, 2024b). 현행 체계에서 택배 수거는 하루 세 차례의 사이클로 운영된다. 각 차량은 매 사이클마다 유일한

본 논문은 인천대학교 2025년도 자체연구비 지원에 의하여 연구되었음(과제번호: 2025-0123). 본 연구는 인천대학교 산업경영공학과 학부전공동아리인 LCIP 3기의 대학생 프로젝트 경진대회 최우수상 출품작 결과물을 추가 보완 및 개선하여 논문화되었으며, 연구 수행 과정에서 인천광역시 물류정책과의 협조를 받음.

<sup>†</sup> 연락저자 : 김병수 교수, 22012 인천광역시 연수구 아카데미로 119(송도동) 인천대학교 산업경영공학과, Tel : 032-835-8482,

Fax : 032-835-0777, E-mail : bskim@inu.ac.kr

2025년 11월 18일 접수; 2026년 1월 23일 수정본 접수; 2026년 2월 19일 게재 확정.

차고지인 부평 거점 집화센터에서 출발하여 수거지를 방문한 후, 수거된 물량을 집화센터에 집화한 후, 부평 거점 집화센터로 복귀한다. 그러나 단일 차고지만을 활용하는 현 방식은 수거지와 차고지 간 불필요한 이동을 야기하여 차량 운행 비용 상승을 초래한다. 이러한 차량 운행 효율 저하는 사업의 지속 가능성뿐만 아니라 추후 서비스 대상 확대라는 사업 목표 달성에도 부정적 영향을 미칠 수 있다. 이를 해결하기 위해서 다양한 위치의 거점 차고지를 추가적으로 운영하여 차량의 불필요한 이동을 줄일 필요가 있다. 본 연구에서는 새로운 거점 차고지를 건설하는 대신, 인천 전역에 분포한 전기차 주차장을 거점 차고지로 활용하는 방안을 제안한다. 이를 통해 차량은 첫 번째 사이클에 부평 거점 집화센터에서 출발한 이후 각 수거 사이클이 종료될 때마다 부평 거점 집화센터를 포함하여 위치상 유리한 거점 차고지에 배정될 수 있으며, 마지막 사이클 종료 시에는 다시 부평 거점 집화센터로 복귀한다. 이러한 방식을 통해 단일 차고지 체계에서 발생하는 비효율적 이동을 최소화하는 것을 목표로 한다. 또한, 본 연구에서는 차량 이동 비용뿐 아니라 전기차 주차장을 거점 차고지로 활용하며 발생하는 차고지 이용 비용과 정해진 사이클 시간을 초과할 경우 발생하는 지연 비용을 함께 고려하여 사업 운영상의 현실적인 제약을 반영한 목적함수를 수립한다. 더 나아가 시 단위의 대규모 문제 상황을 효율적으로 처리하기 위해 군집화-할당-경로 계획 3단계 접근법을 제시하고, 이를 통해 ‘소상공인 택배 지원사업’의 운영 효율 개선 방안을 모색한다.

본 논문의 구성은 다음과 같다. 제2장에서는 다중 차고지 차량 경로계획문제(Multi-Depot Vehicle Routing Problem, MDVRP)과 관련된 기존 연구를 검토하고, 제3장에서는 본 연구에서 제안하는 혼합정수선형계획(Mixed Integer Linear Programming, MILP) 모델을 제시한다. 제4장에서는 문제 해결을 위한 3단계 접근법을 설명하며, 제5장에서는 실험과 분석 결과를 제시한다. 마지막으로 제6장에서는 결론과 향후 연구 과제를 논의한다.

## 2. 선행연구

본 연구에서 다루는 문제는 차량이 단일 차고지에서 출발하여 다른 차고지에서 종료하는 경우, 여러 차고지에서 출발하여 경로상 유리한 차고지에서 종료하는 경우, 그리고 여러 차고지에서 출발하여 동일한 차고지에서 종료하는 경우를 포함한다. 이는 차량이 여러 차고지를 활용하면서 경로를 계획하는 상황으로, Tillman(1969)이 제안한 MDVRP의 변형 문제로 볼 수 있다. 따라서 본 절에서는 MDVRP와 그 변형 문제들을 중심으로 관련 선행연구들을 조사하였다.

MDVRP는 여러 차고지에서 차량이 출발해 고객 노드를 방문한 뒤, 출발한 차고지로 복귀하는 최적 경로를 탐색하는 문제로, 일반적으로 차량의 총 이동 거리 또는 총 운영비용을 최소화하는 것을 목표로 하며 물류, 배달, 쓰레기 수거 등 다양한 분야에서

활용된다. MDVRP 관련 최근 연구 동향은 다음과 같다. Pu *et al.*(2022)은 연료 비용과 탄소 배출 비용을 총 운영비용의 일부로 고려하여 물류비용과 차량 이동 거리 최소화를 위한 MILP 모델을 제시하고, 화학반응 최적화 알고리즘(Chemical Reaction Operator, CRO)을 사용하였다. Gu *et al.*(2022)은 MDVRP에서 총 이동 거리를 최소화하기 위해 차고지 군집화를 통해 단일 차고지 차량경로계획문제로 변환한 후, 단일 차고지에 맞춰 인공 벌 군집 알고리즘(Artificial Bee Colony, ABC)을 적용하여 차고지별 해를 생성하고, 공진화 전략(Coevolution Strategy)을 통해 최종 해를 도출하였다. Li *et al.*(2024)는 시간 제약이 있는 다중 차고지 차량경로계획문제(Multi-Depot Vehicle Routing Problem with Time Window, MDVRPTW)를 해결하기 위해 K-means 군집화 알고리즘을 적용한 후 하이브리드 개미 군집 최적화 알고리즘(Hybrid Ant Colony, HACO)을 적용하는 2단계 접근 방법을 사용하였다. Soares and Roboredo(2024)은 각 차량이 출발한 차고지로 돌아올 필요 없는 개방형 다중 차고지 차량경로계획문제(Multi-Depot Open Vehicle Routing Problem, MDOVRP)를 분기-절단 및 평가해법을 사용하여 최적해를 도출하였다. Li *et al.*(2024)는 다중 차고지 다중 타입 어텐션(Multi-Depot Multi-Type Attention, MD-MTA) 모델을 MDVRP와 MDOVRP에 적용하였다. 또한, 도착 차고지를 유연하게 할당하는 다중 차고지 차량경로계획문제(Multi-Depot Vehicle Routing Problem with Flexible Assignment of End Depot, MDVRFED)와 관련한 연구도 진행되었다. Kek *et al.*(2008)은 MDVRP에서 출발 및 도착 차고지를 유연하게 선택할 수 있는 네트워크 모델과 MILP 모델을 제시하였다. Li *et al.*(2016)은 거리제약과 용량제약이 존재하는 차량경로계획문제(Distance-Constrained Capacitated Vehicle Routing Problem, DCVRP)에서 차고지를 유연하게 할당하는 문제를 해결하기 위해 MILP와 하이브리드 유전 알고리즘(Hybrid Genetic Algorithm, HGA)을 제안하였다. Markov *et al.*(2016)은 차량이 중간 시설을 경유하면서 최적의 종료 차고지를 선택하는 쓰레기 수거 문제(Vehicle Routing Problem with Intermediate Facilities: VRP-IF)를 다중 이웃 탐색 휴리스틱으로 해결하였다. Kraiem *et al.*(2025)에서는 도로 청소 문제에서 각 근무 교대마다 도착 차고지를 유연하게 할당하기 위한 MILP 모델을 제안하였다. <Table 1>은 MDVRP 선행연구들의 특징과 본 연구와 선행연구들과의 차이점을 요약한다.

기존 연구들은 차량의 이동 거리와 운영비용을 최소화하는 것을 주된 목표로 설정하고, MILP, GA, ACO, PSO 등 다양한 최적화 기법을 활용하여 문제를 해결해왔다. 그중 본 연구에서 다루는 문제와 가장 유사한 형태는 차고지의 유연한 할당을 고려한 연구들로, 차량이 출발한 차고지로 복귀할 필요 없이 상황에 따라 최적의 종료 차고지를 선택하는 방식을 다루었다. 그러나 이러한 연구들은 차고지 선택의 유연성만을 반영할 뿐, 특정 차고지를 선택하고 이용함으로써 발생하는 비용을 목적함수에 명시적으로 고려하지 않았다. 또한, 본 연구는 하루에 여러 사이클이 연속적으로 운영되는 문제를 다루는 반면, 기존 연구들은 단일 사이

Table 1. Related literature of MDVRP

Author(s)	Problem Environment	End Depot		Objective Function	Solution Methodology	
		Fixed	Flexible		VRP	Clustering
Pu <i>et al.</i> (2022)	MDVRP	✓	-	fixed vehicle cost, fuel cost, carbon emission cost	CRO	
Gu <i>et al.</i> (2022)	MDVRP	✓	-	travel distance	ABC	✓
Li <i>et al.</i> (2024)	MDVRPTW	✓	-	fixed vehicle cost, travel cost, tardiness cost	HACO	✓
Soares and Roboredo(2024)	MDOVRP	✓	-	travel cost, number of routes	Branch Cut and Price Algorithm	
Li <i>et al.</i> (2024)	MDVRP, MDOVRP	✓	-	route length	MD-MTA	
Kek <i>et al.</i> (2008)	MDVRPFED	-	✓	vehicle utilization cost, travel cost	MILP	
Li <i>et al.</i> (2014)	MDVRPFED, DCVRP	-	✓	travel cost	MILP, HGA	
Markov <i>et al.</i> (2016)	MDVRPFED, VRP-IF	-	✓	fixed vehicle cost, travel cost, relocation cost	Multiple Neighborhood Search Heuristic	
Kraiem <i>et al.</i> (2025)	MDVRPFED	-	✓	travel time	MILP	
This Study	MDVRPFED	-	✓	travel cost, depot operational cost, tardiness cost	Three-stage Heuristic	✓

클 기반의 문제를 다루어 특정 사이클의 지연이 후속 사이클에 미치는 영향을 고려하지 않았다.

따라서 본 연구의 기여는 다음과 같다. 첫째로, 차고지의 유연한 할당을 고려한 기존 연구들이 다루지 않은 차고지 이용 비용과 지연 비용을 목적함수에 고려하였다. 문제의 특성상 도시 내 유료 주차장을 거점 차고지로 활용하는 문제 상황을 다루기 위해서는 차고지 이용 비용을 고려해야 하며, 특정 사이클의 지연이 후속 사이클에 영향을 미쳐 서비스 운영의 안정성을 저해할 수 있으므로 지연 비용 또한 고려해야 한다. 이에 본 연구는 차량 이동 비용, 차고지 이용 비용, 지연 비용을 종합적으로 고려하는 새로운 문제 상황의 택배 수거 경로 계획을 다룬다. 두 번째로, 대규모 경로 계획 문제에 효과적으로 대응하기 위한 군집화-할당-경로 계획의 3단계 접근법을 제시한다. 군집화 단계에서는 본 문제의 운영 제약을 충족하기 위해 Density-Based Spatial Clustering of Applications with Noise(DBSCAN)을 기반으로 변형된 군집화 알고리즘을 적용하고, 경로 계획 단계에서는 모의 담금질 기법(Simulated Annealing, SA)에 군집 할당 조정 알고리즘을 추가하여 초기 군집 할당의 품질에 대한 의존성을 보완하였다.

### 3. 문제 설명 및 수리모형

본 연구에서 다루는 택배 수거 경로 계획 문제는 다음과 같은 운영적 특징을 가진다. 첫째, 부평 거점 집화센터는 차고지의

역할과 동시에 수거된 물량을 집화하는 하차지의 역할을 수행한다. 추가로, 인천 전역에 분포한 전기차 주차장은 거점 차고지로 활용되며, 이 경우 이용 시간에 따른 차고지 이용 비용이 발생한다. 둘째, 모든 차량은 수거를 마친 후 반드시 하나의 하차지에 들러 물량을 하차해야 하며, 이후 차고지로 이동하여 투어를 종료한다. 셋째, 동일 수거지를 방문하더라도 선택되는 하차지와 차고지에 따라 전체 비용 및 경로가 달라져, 경로 설계의 복잡도가 높다. 넷째, 현재 운영되는 차량의 수와 지정된 하차지가 정해져 있다. 마지막으로, 각 수거 사이클의 수요가 사이클 시작 시점에 확정된다. 따라서 본 연구는 하나의 사이클을 독립적인 문제 단위로 정의하여 모형과 해법을 적용한다. 이와 같은 상황을 반영하여 본 연구의 문제에 대한 기본 가정은 다음과 같다. 첫째, 모든 택배 물량은 반드시 수거되어야 한다. 둘째, 각 차량은 동일한 용량 제한을 가지며 수거지 수, 수거지별 물량, 차고지 및 하차지의 수는 고정되어 있다. 셋째, 본 연구에서 다루는 소형 택배의 특성상 건당 서비스 시간이 비교적 짧고 균일하므로, 모형의 간소화를 위해 수거 및 하차 소요 시간은 고려하지 않는다. 마지막으로, 하루에 여러 사이클이 연속적으로 운영되는 본 문제의 특성에 따라, 정해진 투어 시간을 초과할 경우 발생하는 지연을 실질적인 경제적 손실로 간주하여(Baals *et al.*, 2023; Mao *et al.*, 2020), 초과 시간에 비례하는 지연 비용이 발생한다. 따라서 본 연구의 목적함수는 차량 이동 비용, 차고지 이용 비용, 그리고 지연 비용의 합을 최소화하는 것이다. 차량 이동 비용은 차량의 총 이동 거리에 단위 거리당 비용을 곱해 정의한다. 차고지 이용 비용은

차량이 차고지에 도착하여 투어를 종료한 시점부터 사이클 종료 시점까지 머무른 시간에 비례하여 산정된다. 차고지는 이용 비용이 발생하지 않는 부평 거점 집화센터와 이용 비용이 발생하는 거점 차고지로 구분된다. 지연 비용은 차량이 정해진 투어 시간을 초과한 경우 초과분에 비례하여 발생한다. <Figure 1>은 단일 차고지만을 활용하는 기존 상황의 예시를 보여준다. 차량 1과 차량 2는 모두 동일한 차고지 B1에서 투어를 출발한다.

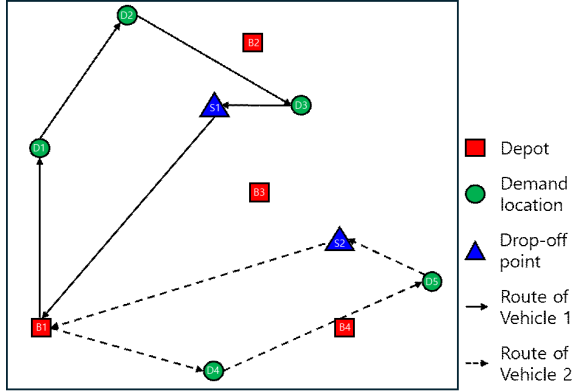


Figure 1. A tour example without the flexible depots strategy

차량 1은 할당된 수거지 D1, D2, D3를 방문하여 택배를 수거한 후, 하차지 S1을 방문하여 택배를 하차한다. 이후 투어를 출발했던 차고지 B1으로 다시 돌아와 투어를 종료한다. 차량 2는 할당된 수거지 D4, D5를 방문하여 택배를 수거한 후, 하차지 S2를 방문하여 택배를 하차한다. 이후 투어를 출발했던 차고지 B1으로 다시 돌아와 투어를 종료한다. <Figure 2>는 거점 차고지를 활용하여 차고지의 유연한 할당을 고려하는 상황의 예시를 보여준다. <Figure 1>의 상황과 다르게 새로운 거점 차고지 B2, B3, B4를 활용하여 차고지가 유연하게 할당된 모습을 확인할 수 있다. 이는 택배 하차 후 다시 출발했던 차고지로 돌아가는 이동 비용보다 가까운 차고지로 들어가 발생하는 차고지 이용 비용이 적어 최종 경로가 달라진다.

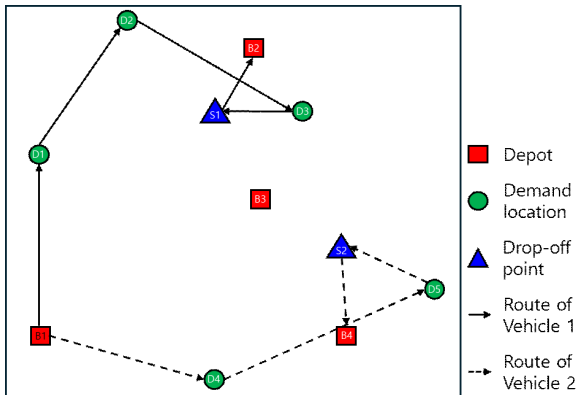


Figure 2. A tour example with the flexible depots strategy

아래는 본 연구의 문제를 해결하기 위해 제시한 MILP 모형이다.

**Sets and Parameters**

- $D$  택배 수거지 집합
- $S$  택배 하차지 집합
- $B$  차량이 이용 가능한 차고지 집합
- $N$  차량이 방문 가능한 모든 노드 집합  $D \cup S \cup B$
- $K$  운영 차량 집합
- $P_k$  각 차량별 출발지,  $\forall k \in K$
- $R_k$  각 차량별 주행 가능 거리,  $\forall k \in K$
- $W$  차량의 적재 가능 용량
- $H$  차량의 투어 가능 시간
- $d_{ij}$  노드 사이 맨해튼 거리,  $\forall i, j \in N$
- $Q_i$  각 택배 수거지별 물량,  $\forall i \in D$
- $T$  단위 거리당 소요 시간
- $DC$  단위 거리당 발생 비용
- $PC_b$  각 차고지별 단위 시간당 차고지 이용 비용,  $\forall b \in B$
- $LC$  단위 시간당 지연 비용
- $n$  수거지와 하차지 노드 수
- $M$  Big M

**Decision Variables**

- $x_{kij}$  차량  $k$ 가 노드  $i$ 에서 노드  $j$ 로 이동하면 1, 아니면 0,  $\forall k \in K, (i, j) \in N$
- $z_{ki}$  차량  $k$ 가 노드  $i$ 를 방문하면 1, 아니면 0,  $\forall k \in K, i \in N$
- $td_k$  차량  $k$ 의 총 이동 거리,  $\forall k \in K$
- $p_{kb}$  차량  $k$ 의 차고지  $b$  이용시간,  $\forall k \in K, b \in B$
- $u_{ki}$  차량  $k$ 가 노드  $i$ 를 방문한 순서,  $\forall k \in K, i \in D \cup S$
- $t_{ki}$  차량  $k$ 가 노드  $i$ 를 방문한 시간,  $\forall k \in K, i \in D \cup S$
- $c_k$  차량  $k$ 가 투어를 종료한 시간,  $\forall k \in K$
- $L_k$  차량  $k$ 의 총 지연 시간,  $\forall k \in K$

목적함수와 제약식은 다음과 같다.

**Objective Function**

$$\begin{aligned} \text{Min} \sum_{k \in K} (td_k \times DC) + \sum_{b \in B} \sum_{k \in K} (p_{kb} \times PC_b) \\ + \sum_{k \in K} (L_k \times LC) \end{aligned} \quad (1)$$

Subject to

$$x_{kii} = 0 \quad \forall k \in K, i \in N \quad (2)$$

$$\sum_{i \in N} x_{kij} = z_{kj} \quad \forall k \in K, i \in D \cup S \quad (3)$$

$$\sum_{j \in N} x_{kij} = z_{ki} \quad \forall k \in K, i \in D \cup S \quad (4)$$

$$z_{kP_k} = 1 \quad \forall k \in K \quad (5)$$

$$\sum_{j \in D} x_{kP_{kj}} = 1 \quad \forall k \in K \quad (6)$$

$$\sum_{k \in K} z_{ki} = 1 \quad \forall i \in D \quad (7)$$

$$\sum_{j \in B} x_{kij} = 0 \quad \forall k \in K, i \in D \quad (8)$$

$$\sum_{i \in S_j \in B} x_{kij} = 1 \quad \forall k \in K \quad (9)$$

$$\sum_{i \in S} x_{kij} = z_{kj} \quad \forall k \in K, j \in B \setminus P_k \quad (10)$$

$$\sum_{i \in B_j \in N} x_{kij} = 1 \quad \forall k \in K \quad (11)$$

$$u_{ki} - u_{kj} + n \times x_{kij} \leq n - 1 \quad \forall k \in K, (i, j) \in \mathcal{MA} \quad (12)$$

$$\sum_{i \in N} z_{ki} \times Q_i \leq W \quad \forall k \in K \quad (13)$$

$$\sum_{i \in N_j \in N} x_{kij} \times d_{ij} \leq td_k \quad \forall k \in K \quad (14)$$

$$td_k \leq R_k \quad \forall k \in K \quad (15)$$

$$t_{ki} \leq T \times d_{P_kj} + M \times (1 - x_{kP_{kj}}) \quad \forall k \in K, j \in D \quad (16)$$

$$T \times d_{P_kj} \leq t_{kj} + M \times (1 - x_{kP_{kj}}) \quad \forall k \in K, j \in D \quad (17)$$

$$t_{kj} \leq t_{ki} + T \times d_{ij} + M \times (1 - x_{kij}) \quad \forall k \in K, i \in D \cup S, j \in D \cup S \quad (18)$$

$$t_{ki} + T \times d_{ij} \leq t_{kj} + M \times (1 - x_{kij}) \quad \forall k \in K, i \in D \cup S, j \in D \cup S \quad (19)$$

$$e_k \leq t_{ki} + \sum_{b \in B} T \times d_{ib} \times x_{kib} + M \times (1 - z_{ki}) \quad \forall k \in K, i \in S \quad (20)$$

$$t_{ki} + \sum_{b \in B} T \times d_{ib} \times x_{kib} \leq e_k + M \times (1 - z_{ki}) \quad \forall k \in K, i \in S \quad (21)$$

$$H - e_k - M \times (1 - \sum_{i \in N} x_{kib}) \leq p_{kb} \quad \forall b \in B, k \in K \quad (22)$$

$$L_k \leq H - e_k \quad \forall k \in K \quad (23)$$

목적함수 (1)은 모든 차량의 이동 비용, 차고지 이용 비용, 지연 비용의 합을 최소화한다. 제약식 (2)는 동일 노드에서 동시에 들어오고 나가는 경로를 방지한다. 제약식 (3), (4)는 각 차량이 수거지나 하차지를 방문했을 때 반드시 하나의 유입 아크와 하나의 유출 아크가 존재해야 함을 보장한다. 제약식 (5), (6)은 각 차량이 반드시 지정된 출발지에서 출발해야 하며, 출발 후에는 반드시 수거지 노드로만 이동해야 한다는 제약이다. 제약식 (7), (8)은 모든 수거지가 반드시 방문 되어야 하며, 수거지에서 하차지를 거치지 않고 바로 차고지로 이동할 수 없음을 나타낸다. 제약식 (9), (10)은 각 차량이 하차지에서 차고지로 이동해야 함을 보장한다. 제약식 (11)은 차량이 차고지에 도착하면 다시 출발할 수 없도록 한다. 제약식 (12)는 서브투어를 방지하는 제약이다. 제약식 (13)은 각 차량의 용량 제한을 나타내며, 제약식 (14)는 차량 이동 관련 비용을 정의한

다. 제약식 (15)은 차량의 최대 이동 가능 거리를 제한한다. 제약식 (16)~(21)은 각 차량이 첫 방문 노드, 경유 노드, 종료 노드에서의 방문 시간을 정의한다. 제약식 (22)는 각 차량의 차고지 이용 시간을 나타내며, 제약식 (23)은 각 차량의 지연 시간을 정의한다.

#### 4. 방법론

본 연구의 3단계 접근법은 다음과 같이 구성된다. 첫째, 수거지를 군집화한다. 둘째, 형성된 군집들을 차량에 할당한다. 셋째, 각 차량에 할당된 군집들에 속하는 수거지에 대해 경로 계획을 수행하고, 군집 할당을 조정해가며 해를 개선한다. 본 장에서는 각 단계별 방법에 대해 상세히 설명한다.

Table 2. Description of parameters

Stage	Parameter	Description
1	$\varepsilon$	The radius used to define the neighborhood of a point
	MinPts	The minimum number of points required to form a dense region
2	$p_0$	Initial acceptance probability of inferior solutions
	$\alpha$	Cooling rate
	$n$	Number of drop and depot points for route optimization
3	$w$	Number of candidate clusters for reallocation
	$\eta$	Minimum improvement threshold

##### 4.1 군집화 알고리즘

VRP는 대표적인 NP-hard 문제로, 대규모 인스턴스의 최적해를 구하는 것은 계산 복잡도가 매우 높다. 이를 해결하기 위한 방법으로 고객 노드를 먼저 군집화한 뒤 군집 단위로 차량에 할당하고 경로를 구성하는 Cluster-First, Route-Second(CFRS) 접근법이 제안되었다(Gillett and Miller, 1974; Fisher and Jaikumar, 1981). CFRS는 문제의 차원을 효과적으로 축소하여 계산 부담을 줄일 수 있으며, 최근 연구에서도 여전히 유효한 접근법으로 평가된다(Zhang *et al.*, 2025). 본 연구 역시 CFRS 프레임워크를 기반으로 군집화 단계를 도입하여 문제의 효율적 해결을 도모하였다. 본 연구의 군집화 단계에서는 DBSCAN을 채택하였다. DBSCAN은 밀도 기반 군집화 알고리즘으로, 사전에 군집 개수를 지정할 필요가 없으며, 비구형(non-spherical) 구조나 불균등하게 분포된 데이터에서도 안정적으로 군집을 형성할 수 있다(Ester *et al.*, 1996; Schubert *et al.*, 2017). 실제 도시 물류 수거지는 균등하게 분포되지 않고 특정 지역에 집중되거나 불규칙하게 분포하는 경향이 있다(Crainic and Toulouse, 2005). 따라서 DBSCAN의 특성은 도시 물류 데이터 다루는 본 연구의 상황에 적합하다. 이러한 선

정의 타당성을 검증하기 위해 군집 개수를 데이터 분포에 따라 자동으로 결정하는 대표적인 알고리즘인 Mean Shift, Affinity Propagation과의 비교 실험을 수행하였다. 실험 결과, DBSCAN이 모든 문제 크기에서 가장 낮은 군집 내 이동 거리와 일관된 분산을 보이며 가장 강건한 군집화 품질을 보였다. 상세 실험 결과는 부록 A에 수록하였다.

DBSCAN의 핵심 파라미터는 이웃 반경의 크기인  $\epsilon$ 과 최소 이웃 수 MinPts다. 본 연구는 물량 발생에 따라 데이터 분포 특성이 유동적으로 변하기 때문에, 문제 상황마다  $k$ -distance 그래프를 기반으로 데이터 분포에 따라 최적  $\epsilon$  값을 동적으로 산정하였다(Beyer *et al.*, 1999). 이를 위해 Kneedle 방법(Satopaa *et al.*, 2011)을 적용하였으며, 해당 방법을 통해 각 노드의 MinPts 번째 최근접 이웃 거리를 계산한 뒤 오름차순으로 정렬하여 얻은  $k$ -distance 곡선에서 기울기의 변화가 급격히 발생하는 지점을 탐지하여 최적  $\epsilon$ 을 도출하였다. MinPts는 본 연구의 데이터 특성에 맞게 값을 변화시키며 성능 차이를 실험적으로 검증하여 파라미터 변화가 군집 수와 결과 품질에 미치는 영향을 체계적으로 분석하고자 한다. 다만, DBSCAN은 기본적으로 일부 노드를 군집에 속하지 않는 노이즈 포인트로 분류한다. 그러나 본 연구의 문제 상황에서는 모든 수거지가 반드시 차량에 의해 방문되어야 한다는 운영상의 제약이 존재한다. 따라서 운영상의 제약 충족을 위해 노이즈 포인트는 가장 가까운 군집에 강제 할당하는 후처리 과정을 적용하였다. 또한, 각 차량에는 최대 적재 용량제약이 존재하므로, 특정 군집의 총 수요가 이를 초과하는 경우  $\epsilon$  값을 점진적으로 축소하여 재분할하는 절차를 추가하여 모든 군집이 차량 적재 용량제약을 만족할 수 있도록 하였다.

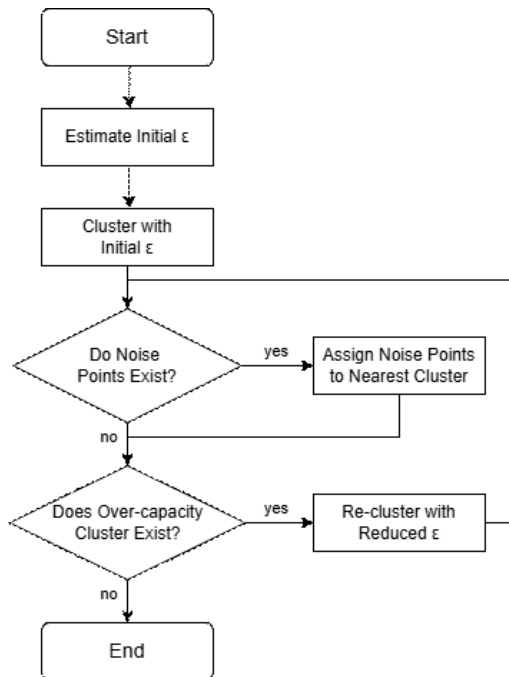


Figure 3. Flowchart for the clustering algorithm

알고리즘의 구체적인 작동 과정은 <Figure 3>과 같다. 먼저,  $\epsilon$  값을 추정된 뒤 DBSCAN을 적용하여 초기 군집을 형성한다. 이후 노이즈 포인트가 존재하면 이를 가장 가까운 군집에 할당한다. 다음으로, 군집의 총 수요가 차량 용량을 초과할 경우 해당 군집에 대해  $\epsilon$  값을 축소하여 다시 군집화를 수행한다. 이때 새롭게 발생한 노이즈 포인트는 동일한 방식으로 가장 가까운 군집에 할당된다. 이러한 절차는 모든 군집이 차량 용량제약을 충족할 때까지 반복된다. 요약하면, 본 연구의 군집화 절차는 (1) DBSCAN을 활용한 초기 군집 형성, (2) 노이즈 포인트의 재할당, (3) 차량 용량을 초과하는 군집 재분할의 세 단계로 구성된다. 이러한 과정은 문제의 차원을 축소하면서도, 기존 접근법을 본 문제 상황에 맞게 변형하여 모든 수거지를 반드시 방문해야 하는 제약과 차량 용량제약을 동시에 반영한다.

#### 4.2 차량-군집 할당

군집화 단계에서 형성된 군집은 여전히 특정 차량에 할당되지 않은 상태이므로, 이를 실제 차량 경로 계획으로 연결하기 위해서는 각 군집을 차량에 효율적으로 할당하는 절차가 요구된다. 본 연구에서는 군집 할당 문제를 혼합정수계획법(Mixed Integer Programming, MIP)을 통해 해결하였다. 군집 할당의 주요 목적함수는 차량별 출발지와 군집 중심 간의 거리를 최소화하는 동시에, 차량 적재 용량 및 고정된 차량 운영 대수와 같은 운영 제약 조건을 충족하는 것이다. 즉, MIP 기반 할당은 단순히 군집화와 차량 경로 계획을 연결하는 절차를 넘어, 차량 운행의 현실적 제약을 사전에 반영하고 후속 경로 계획의 효율성을 높이는 역할을 한다.

#### Sets and Parameters

- $D$  군집 중심점 노드 집합
- $B$  차량이 이용 가능한 차고지 집합
- $N$  차량이 방문 가능한 모든 노드 집합  $D \cup B$
- $K$  운영 차량 집합
- $P_k$  각 차량별 출발지,  $\forall k \in K$
- $W$  차량의 적재 가능 용량
- $d_{ij}$  노드 사이 맨해튼 거리,  $\forall i, j \in N$
- $Q_i$  각 군집별 물량,  $\forall i \in D$
- $m$  각 차량의 최소 군집 할당 개수

#### Decision Variables

- $z_{ki}$  차량  $k$ 가 군집  $i$ 를 할당받으면 1, 아니면 0,  
 $\forall k \in K, i \in N$

#### Objective Function

$$\text{Min} \sum_{k \in K} \sum_{i \in D} (d_{p_k i} \times z_{ki}) \quad (24)$$

Subject to

$$\sum_{k \in K} z_{ki} = 1 \quad \forall i \in D \quad (25)$$

$$\sum_{i \in D} z_{ki} \times Q_i \leq W \quad \forall k \in K \quad (26)$$

$$\sum_{i \in D} z_{ki} \geq m \quad \forall k \in K \quad (27)$$

목적함수 (24)는 각 차량의 출발지와 할당된 군집 중심 간 거리의 합을 최소화한다. 제약식 (25)는 모든 군집이 반드시 할당되어야 함을 보장하며, 제약식 (26)은 차량의 적재 용량제약을 반영한다. 마지막으로 제약식 (27)은 모든 차량이 최소한 한 개 이상의 군집을 할당받아 운영되어야 함을 의미한다.

### 4.3 경로 계획 알고리즘

본 연구는 SA와 군집 할당 조정 알고리즘을 결합한 경로 계획 알고리즘을 제안한다. 앞서 설명한 과정을 통해 군집이 각 차량에 할당되면, 각 차량이 방문해야 할 수거지 노드가 모두 정해지게 된다. 이후, 각 차량별로 메타휴리스틱 알고리즘을 활용해 목적함수 값을 도출하여 전체적인 군집 할당 결과를 평가하고, 차량 간 군집 이동을 통해 총 목적함수 값을 개선하여 차량별 최종 경로와 최종 목적함수 값을 도출한다.

#### (1) 모의 담금질 기법

본 연구에서는 대규모 문제 상황에서 다수의 차량에 대해 반복적으로 방문 순서를 탐색해야 하기 때문에 대규모 조합 최적화 문제에서 온도에 따라 목적함수 값이 증가하더라도 해당 해를 수용하여 지역 최적해에 빠지지 않게 하고, 합리적인 시간 내에 양질의 근사해를 탐색할 수 있는 메타휴리스틱 기법으로 알려진 SA를 사용한다(Kirkpatrick *et al.*, 1983). 해당 방법이 적용되는 상황에서는 각 차량이 방문해야 할 수거지가 모두 정해져 있기 때문에 노드 방문 순서만을 결정하는 문제를 고려한다. 따라서, 해는 차량별 방문해야 할 노드들을 나타내는 숫자들의 순열로 정의한다. 해의 마지막 두 부분은 순서대로 차량이 방문할 하차지, 차고지로 고정한다. 목적함수는 식 (1)을 그대로 사용한다. 해당 과정의 자세한 내용은 <Figure 4>에 설명되어 있다.

우선, 초기 해  $x_0$ 는 노드 간 맨해튼 거리를 기준으로 최근접 이웃 탐색을 통해 형성한다. 이후, 인접해 탐색 방법으로는 외판원 문제와 같은 순열형 경로 문제에서 전통적으로 활용되어 온 부분 수열 역전을 사용한다(Kirkpatrick *et al.*, 1983). 해당 방법은 임의의 두 인덱스를 선택해 그 사이 구간을 역전시키는 방법이다. 인접해 탐색 과정에서 하차지와 차고지는 해의 마지막 부분에 순서대로 위치해야 하기 때문에 부분 수열 역전 방법은 수거지 부분에만 적용하고, 하차지는 마지막 수거지에서 가까운  $n$ 개의 하차지중 하나를 무작위로 선택하고, 차고지도 같은 방식으로 하차지에서 가까운  $n$ 개의 차고지 중에 무작

위로 선택한다. 해당 과정은 Figure 4에  $tweak(x;n)$ 으로 표기되어 있다. 이웃 탐색 과정에서 지연 시간이 크게 발생하게 되면 비용이 급격하게 증가하기 때문에, 특정 시간  $\tau$ 보다 지연 시간이 크게 발생한다면 탐색 과정에서 해당 해는 수용하지 않는다.

어떤 해  $x$ 의 목적함수 값을  $f(x)$ , 이웃 탐색 과정 후의 해  $x'$ 의 목적함수 값을  $f(x')$ 라고 했을 때, 두 목적함수 값의 차이를  $\Delta f = f(x') - f(x)$ 라고 하면, 탐색 수용 확률은 시점별 온도  $T_t$ 에 따라  $e^{-\Delta f/T(t)}$ 로 정의된다. 0과 1 사이 무작위 난수 값  $r$ 을 생성하여  $r < e^{-\Delta f/T(t)}$ 라면 해당 해를 수용하고 그렇지 않다면 수용하지 않는다. 이후 온도는 감가율  $\alpha$ 에 따라  $T_{t+1} = \alpha T_t$ 로 감소하며, 온도가 최소 온도  $\omega$ 에 도달할 때까지 탐색 과정을 반복한다. 따라서 SA의 파라미터인 초기 온도  $T_0$ 는 탐색 정도를 결정하기 때문에 알고리즘 성능에 큰 영향을 준다(Ben-Ameur, 2004). 본 연구에서는 각 차량별로 할당 받은 클러스터에 따라 다양한 문제 상황이 발생하기 때문에 특정 초기온도를 모든 문제 상황에 적용하기보다, 각 차량별 문제 상황에 따라 유동적으로 초기온도를 설정할 수 있는 방법을 사용한다. 본 연구는 각 차량에 대해 Ben-Ameur(2004)의 방법을 따라 일정한 목표 초기 열등해 수용 확률  $p_0$ 를 만족하는  $T_0$ 를 추정한다. 우선 초기 해  $x_0$ 의 비용  $f(x_0)$ 을 구하고, 인접해 탐색을 반복적으로 수행하여 비용이 증가하는 이동들로 구성된  $m$ 개의 표본을 수집한 뒤, 온도  $T$ 에서의 평균 수용 확률을 식 (28)과 같이 추정한다.

```

Input :  $\alpha, n, \omega, \tau$ 
Initialize  $T \leftarrow T_0, x \leftarrow x_0, x^* \leftarrow x_0$ 
while  $T \geq \omega$  do
   $x' = tweak(x;n)$ 
  if arrival time  $> \tau$  then
     $T \leftarrow \alpha T$ 
  continue
  end if
   $\Delta f \leftarrow f(x') - f(x)$ 
  if  $\Delta f \leq 0$  then
     $x \leftarrow x'$ 
  else
     $r = random()$ 
    if  $r < e^{-\Delta f/T}$  then
       $x \leftarrow x'$ 
    end if
  end if
  if  $f(x) < f(x^*)$  then
     $x^* \leftarrow x$ 
  end if
   $T \leftarrow \alpha T$ 
end while
return  $f(x^*)$ 

```

Figure 4. Pseudo-code for the SA algorithm

$$\hat{\chi}(T) = \frac{\sum_{i=1}^m e^{-f(x_i)/T}}{\sum_{i=1}^m e^{-f(x)/T}} \quad (28)$$

이후 평균 수용 확률을  $p_0$ 에 맞추도록  $T_{k+1} = T_k \left( \frac{\ln \hat{\chi}(T_k)}{\ln p_0} \right)^{1/p}$ 로 반복적으로 보정한다. 위 과정을 통해 차량별로 동일한 수용 확률을 충족시킬 초기온도를 설정할 수 있다.

## (2) 군집 할당 조정 알고리즘

4.2절에서 설명한 할당 방법은 단순히 거리만을 고려하여 각 차량에 군집을 할당하기 때문에 본 문제의 목적함수를 반영하여 차량-군집 할당을 개선하기 위한 군집 할당 조정 알고

```

Input :  $V, C, C_v, g_c, g_v, x_v, d(\cdot, \cdot), w, f(\cdot), \eta$ 
Initialize Improved  $\leftarrow$  false
repeat
Improved  $\leftarrow$  false
 $V' \leftarrow \text{argsort}_{v \in V}(f(x_v), \text{descend})$ 
for  $i = 1, \dots, |V|$  do
 $\hat{v} \leftarrow V_i$ 
if  $|C_{\hat{v}}| \leq 1$  then
continue
end if
 $g_{\hat{v}} \leftarrow \frac{1}{|C_{\hat{v}}|} \sum_{c \in C_{\hat{v}}} g_c$ 
 $\hat{c} \leftarrow \text{argmax}_{c \in C_{\hat{v}}} d(g_c, g_{\hat{v}})$ 
 $\sigma \leftarrow \text{argsort}_{c \in C_{\hat{v}}} (d(g_c, g_{\hat{c}}), \text{ascend})$ 
 $N \leftarrow (\sigma_1, \dots, \sigma_w)$ 
 $U \leftarrow \{u \in V \setminus \{\hat{v}\} : C_u \cap N \neq \emptyset\}$ 
for  $u = 1, \dots, |U|$  do
 $F \leftarrow f(x_{\hat{v}}) + f(x_u)$ 
 $C_{\hat{v}} \leftarrow C_{\hat{v}} \setminus \{\hat{c}\}$ 
 $C_u \leftarrow C_u \cup \hat{c}$ 
 $F' \leftarrow f(x_{\hat{v}}) + f(x_u)$ 
 $\rho = \frac{F - F'}{F}$ 
if  $\rho \geq \eta$  then
Improved  $\leftarrow$  True
break
else
 $C_u \leftarrow C_u \setminus \{\hat{c}\}$ 
 $C_{\hat{v}} \leftarrow C_{\hat{v}} \cup \{\hat{c}\}$ 
end if
end for
end for
until Improved = false

```

Figure 5. Pseudo-code for the Adjustment Algorithm

리즘을 제안한다. 4.3.1절에서 설명한 SA 알고리즘을 활용하여 차량별 목적함수 값을 도출하고, 해당 값에 기반해 군집을 다른 차량으로 이동시키는 과정을 반복적으로 수행하며 군집 할당을 조정함으로써 목적함수 값을 개선한다. 해당 과정의 자세한 내용은 <Figure 5>에 설명되어 있다.

우선, 전체 차량 집합  $V$ 에 대해 알고리즘을 통해 구한 차량  $v \in V$ 의 최종 해  $x_v$ 에 대해 SA를 통해 구한 목적함수 값  $f(x_v)$ 를 기준으로 차량을 내림차순으로 정렬한다.

이후, 가장 비용이 큰 차량부터 후보 차량  $\hat{v}$ 로 정의하고 군집 이동을 시도한다. 전체 군집 집합  $C$ 중 후보 차량  $\hat{v}$ 에 할당되어있는 군집 집합  $C_{\hat{v}}$ 와 각 군집  $c \in C_{\hat{v}}$ 의 중심점  $g_c$ 에 대해 다른 차량으로 이동시킬 군집  $\hat{c}$ 는 다음과 같이 하나의 차량  $\hat{v}$ 에 할당된 군집의 중심점  $g_{\hat{v}}$ 를 기준으로, 중심점 간 맨해튼 거리  $d(g_c, g_{\hat{v}})$ 가 가장 큰 군집으로 결정한다. 식 (29)는 차량의 중심점과 이동 후보 군집을 결정하는 식이다.

$$g_{\hat{v}} = \frac{1}{|C_{\hat{v}}|} \sum_{c \in C_{\hat{v}}} g_c, \quad \hat{c} = \text{argmax}_{c \in C_{\hat{v}}} d(g_c, g_{\hat{v}}) \quad (29)$$

이후, 해당 군집의 중심점을 기준으로 가까운 곳에 위치한  $w$ 개의 군집을 선택한다.  $w$ 개의 군집 중 각 군집이 할당된 차량을 최종 군집 이동 후보 차량 집합  $U$ 로 선정하고, 차량  $u \in U$ 로 군집  $\hat{c}$ 를 이동시킨 뒤 각 차량의 목적함수 값  $f(x_{\hat{v}})$ ,  $f(x_u)$ 에 대해 다음식 (30)과 같이 개선율  $\rho$ 를 구한다.

$$\rho = \frac{f(x_{\hat{v}}) + f(x_u) - (f(x_{\hat{c}}) + f(x_u))}{f(x_{\hat{v}}) + f(x_u)}, \quad u \in U \quad (30)$$

$\rho$ 가 임의의 임계값  $\eta$ 보다 크거나 같은 경우에만 군집 이동을 수락하고, 그렇지 않은 경우에는 다음 이동 후보 차량에 대해 이동을 시도한다. 해당 과정을 반복하여 하나의 차량에 대해 개선이 되면, 다시 전체 차량을 목적함수 값을 기준으로 정렬한 후 앞서 설명한 과정을 반복한다. 만약 개선에 실패했다면, 그 다음 차량을  $\hat{v}$ 로 정해서 위 과정을 반복한다. 모든 차량에 대해 시도했음에도 이동이 이루어지지 않는다면, 해당 알고리즘은 종료되고 최종 경로와 목적함수 값을 도출하게 된다. 해당 과정을 통해  $g_{\hat{v}}$ 를 기준으로 외곽에 위치하는 군집을 인접한 차량에 이동시키며 개선된 경로를 탐색할 수 있게 된다.

해당 과정은 4.1절에서 설명한 군집화 알고리즘에서 MinPts에 의해 결정되는 군집 개수에 의존적이다. 군집 개수가 많고 군집 하나에 포함된 수요지가 적다면, 군집 이동 알고리즘을 통해 세밀한 조정이 가능하지만, 연산 시간이 증가하게 될 것이다. 반대로, 군집 개수가 너무 적고 크기가 크다면 세밀한 조정 과정을 거치지 못하게 될 것이다. 이 부분은 다음 5장에서 MinPts에 따른 실험을 통해 확인하고자 한다.

## 5. 실험

본 장에서는 3장에서 개발한 MILP에 대해 CPLEX solver를 활용해 도출한 최적해와 4장에서 제시한 알고리즘의 해를 비교하여, 본 연구의 방법론의 최적해 근사 능력을 검증한다. 이후 현실적인 크기의 문제에 대해 문제 크기와 DBSCAN의 파라미터인 MinPts를 변화시키며 진행한 실험을 통해 데이터 규모와 군집 구조의 변화가 해의 품질과 연산 시간에 미치는 영향을 분석한다. 마지막으로 본 연구의 방법론과 전략의 효과를 검증한다. 구체적으로, 군집화 도입 여부를 비교하여 군집화 절차의 효율성을 평가하고, 차고지의 유연한 할당을 고려하는 전략의 도입 여부를 비교하여 도시 내 전기차 주차장을 추가적인 거점 차고지로 활용하는 방안이 사업 운영비용 절감에 미치는 효과를 분석한다.

### 5.1 실험 계획

작은 크기의 문제는 MILP가 제한 시간 내에 풀 수 있는 범위를 고려하여 설정하였다. 운영 차량 수는 2대, 하차지 수(S)와 차고지 수(B)의 조합을 (3,4)와 (6,8)로 설정하였다. 작은 크기의 문제에서는 클러스터 할당 조정을 위한 클러스터 수를 확보하기 위해 MinPts는 2로 설정하였으며 수거지 수(D)를 10, 20, 30, 40으로 변화시켜 총 8개의 인스턴스를 생성하였다. 이후 각 문제에 대해 CPLEX solver로 도출한 해와 본 연구의 방법론으로 도출한 해를 비교한다. CPLEX solver의 해는 최적해 또는 시간제한 내에 도출된 best solution을 포함한다. 각 사이클의 수요가 사이클 시작 시점에 확정되는 운영 특성상 경로 계획에 장시간을 할애하기 어려우므로, 시간제한은 3,600초로 설정하였다. 식 (31)에 따라 Gap과 CPU 시간을 비교하여 본 연구의 방법론의 최적해 대비 근사 정확도와 계산 효율을 검증한다.

$$\text{Gap}(\%) = \frac{\text{Three-stage Heuristic Algorithm Solution} - \text{CPLEX Solution}}{\text{CPLEX Solution}} \times 100 \quad (31)$$

현실적인 크기의 문제는 사업 운영 담당자 인터뷰를 참고하여 운영 차량 30대, 지정된 하차지 15곳, 수거지 수는 1,200, 1,400, 1,600, 1,800개로 설정하였다. 추가적인 거점 차고지는 인천광역시시의 주차 공간 중 주차 자리가 25대 이상 확보된 비거주지 개방형 전기차 주차장 15곳을 선정하여 활용하였다. 하차지와 차고지의 위치는 <Figure 6>과 같다. 또한, 군집의 개수와 구조를 조절하여 문제 크기와 군집화 결과가 해 품질과 연산 시간에 미치는 영향을 비교 분석하기 위해 MinPts를 변화시켜 실험을 수행하였다. MinPts를 6 이상으로 설정할 경우 군집 수가 운영 차량 대수(30대) 미만으로 형성되는 경우가 발생하여 MinPts는 2, 3, 4, 5 범위를 채택하였으며, 이를 통해 총 16개의 인스턴스를 생성하였다. 수거지의 위치는 인천광역시

지역구별 통신판매업자 주소 데이터를 기반으로 생성하였으며 자세한 출처는 부록 B에 수록하였다.

모든 문제의 세부 파라미터는 다음과 같다. 각 수거지별 물량은 도시 물류 데이터 기반으로 [1, 10] 범위의 균등분포에서 무작위로 발생하도록 설정하여 평균 5.5개의 물량이 생성되도록 하였다(Robichet *et al.*, 2022). 사업 운영 담당자 인터뷰를 참고하여 차량의 적재 한도는 400개, 차량의 투어 가능 시간은 180분으로 설정하였다. 본 연구에서 노드 간 거리는 맨해튼 거리로 산정하였다. 차량의 주행 가능 거리는 211km(Hyundai Motor Company, 2025), 단위 거리당 비용은 90원/km(Korea Energy Agency, 2025), 단위 거리당 소요 시간은 2.8분/km로 환산하였으며(Statistics Korea, 2025), 부평 집화센터를 제외한 거점 차고지 이용 비용에는 회차 시간 이후 시간당 1,020원을 부과하였다(Incheon Facilities Corporation, 2025). 또한, 연속적인 사이클 운행으로 인해 지연 발생을 최소화해야 하는 운영상의 요구를 모형에 반영하고자, 지연 비용 스케일에 따른 민감도 분석을 통해 도출된 값을 지연에 따른 페널티 비용으로 설정하였다. 본 연구에서 SA의 초기 열등해 수용 확률  $p_0$ 와 감가율  $\alpha$ 는 실험적으로 각각 0.5와 0.995로 설정하였으며 파라미터별 실험 결과는 부록 C에 수록하였다. 또한, 인접해 탐색에서 차고지와 하차지 선택에 사용되는 파라미터인  $n$ 은 하차지와 차고지의 개수 및 위치를 고려하여 5로 설정하였다. 또한, 군집 할당 조정 알고리즘에서 최소 개선율  $\eta$ 는 지나치게 미세한 개선으로 인한 계산 낭비를 줄이기 위해 1%로 설정하였고, 군집 이동을 위해 탐색하는 근접 클러스터 수  $w$ 는 탐색 효율성을 높이기 위해 전체 클러스터 수의 10%로 제한하였다. Solver는 IBM ILOG CPLEX 22.1.1을 사용하였으며, 알고리즘은 파이썬 프로그래밍 언어로 구현되었다. 이 실험은 AMD Ryzen 5 7500F, RAM 32GB의 PC 환경에서 진행되었다.

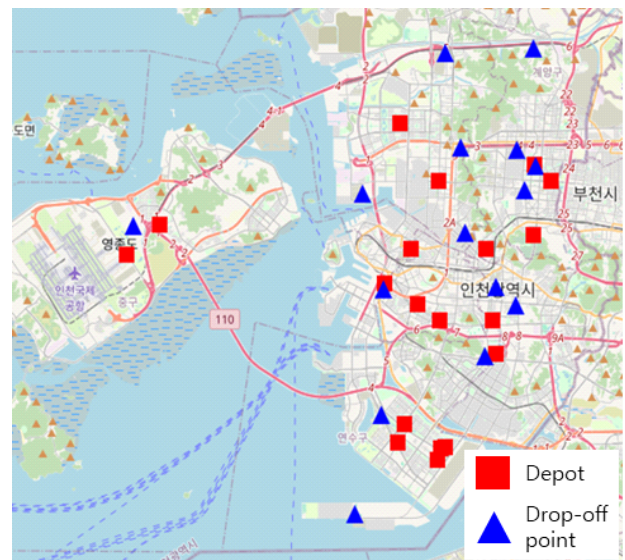


Figure 6. A geographical distribution of drop-off points and depots

5.2 실험 결과

<Table 3>은 작은 크기의 문제에 대해 CPLEX로 도출한 해와 본 연구의 방법론으로 10회 반복실험하여 도출한 해의 평균값 간의 차이를 Gap으로 나타내었다. CPLEX가 제한된 시간 내에 최적해를 도출하지 못한 경우는 N/A로 표기하였으며, 이 경우 Gap은 시간제한 내 best solution과의 비교 결과이다. 최적해가 도출된 인스턴스(1~3, 5~7)에서 본 연구의 방법론은 최대 5.85%의 Gap을 보이며 근사해를 효과적으로 도출하는 것을 확인하였다. 한편, 인스턴스 4와 8에서는 CPLEX가 제한된 시간 내에 최적해를 도출하지 못하였으며, 이 경우 본 연구의 방법론이 best solution 대비 각각 16.60%, 15.17% 낮은 비용의 해를 도출하였다. CPLEX는 수거지 수가 증가함에 따라 연산 시간이 급격하게 늘어나, 수거지 40개 수준에서도 제한된 시간 내에 최적해를 도출하지 못하였다. 본 연구에서 고려하는 문제는 수거지 1,200개 이상의 대규모 문제이므로, 현실적인 시간 내에 양질의 해를 도출할 수 있는 본 연구의 방법론이 요구

된다.

현실적인 크기에 대한 실험을 진행하기에 앞서, 지연 비용의 스케일에 따른 민감도 분석을 통해 적절한 지연 비용을 설정하고자 한다. 시간당 지연 비용을 주차장 시간당 주차장 이용 비용인 1,020원의 약 10배 수준인 10,000원까지 등 간격으로  $LC = \{0, 2500, 5000, 7500, 10000\}$  범위를 설정하여 실험을 진행하였다. <Figure 7>은 가장 큰 문제 사이즈인  $|D|=1,800$ 에 대한 실험 결과로, 왼쪽의 비용 축을 따르는 차량 이동 비용과 차고지 이용 비용의 합과 오른쪽의 시간 축을 따르는 지연 시간으로 구성되어 있다. 지연 비용이 증가함에 따라 총 지연 시간은 약 12.7% 단축되었다. 또한, 차량 이동 비용과 차고지 이용 비용의 합은 지연 비용 증가에 따라 0.75% 정도의 미세한 감소를 보였으며, 이는 지연 비용을 키우는 것이 다른 비용 항목들을 잠식하지 않으면서도 알고리즘이 시스템 운영의 정시성을 고려하도록 유도할 수 있음을 보여준다. 따라서, 이후 모든 실험에서는 지연 시간이 가장 적게 나타날 수 있도록 시간당 지연 비용을 10,000원으로 설정하였다.

Table 3. Testing Results of Small-sized Instances

Ins.	S	B	D	CPLEX		Three-stage Heuristic Algorithm	
				Opt. Cost(₩)	CPU(s)	Gap(%)	CPU(s)
1	3	4	10	4,198	0.8	2.57	4.5
2			20	5,141	1.6	3.83	11.0
3			30	5,688	127.0	5.85	7.4
4			40	N/A	3,600+	-16.60	8.9
5	6	8	10	4,205	2.2	2.59	5.4
6			20	5,735	14.6	1.90	6.4
7			30	5,694	274.0	5.85	7.8
8			40	N/A	3,600+	-15.17	10.26

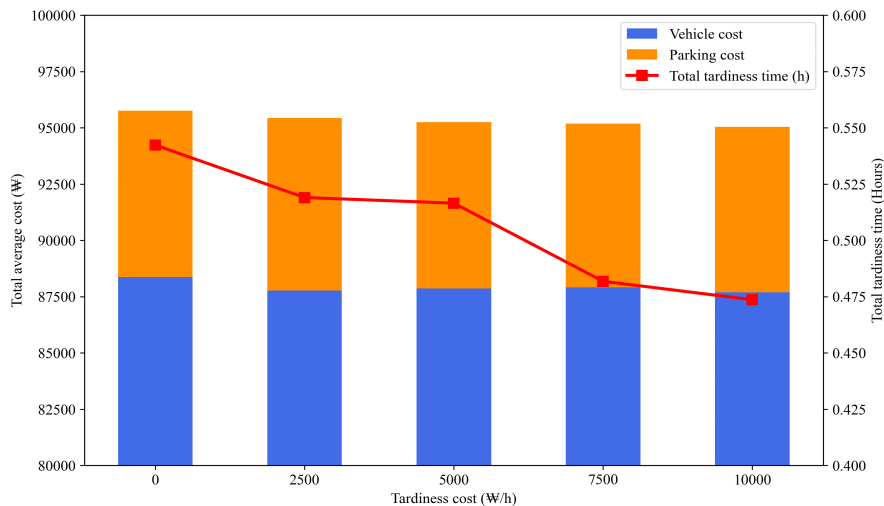


Figure 7. Sensitivity analysis of total average cost and tardiness time to tardiness Cost

<Table 4>는 현실적인 크기에 대한 문제의 실험 결과이다. Average Cost는 10회 반복실험의 평균 목적함수 값이며, Initial Cost는 차량-군집 할당 과정 후 경로 계획 알고리즘이 적용되기 전의 거리를 기준으로 한 최근접 이웃 탐색 방법으로 경로를 설정했을 때의 초기 해를 의미하며, Final Cost는 최종 해를 의미한다. 실험 결과, 모든 인스턴스에서 최종 해는 초기 해보다 목적함수 값이 낮게 나타났다. 이는 제안한 방법론이 단순 초기 해 대비 안정적인 개선 효과를 제공함을 보여주며, 문제 규모와 MinPts 값과 관계없이 일정 수준 이상의 효율성을 확보할 수 있음을 시사한다. 또한, 문제 규모에 따라 적합한 MinPts 값이 서로 다르게 나타났다.  $|D|=1,200$ 은 MinPts=5에서,  $|D|=1,400$ 은 MinPts = 2,  $|D|=1,600$ 과  $|D|=1,800$ 은 MinPts=3에서 가장 좋은 결과가 확인되었다. 이는 단일한 MinPts 값으로 모든 문제에서 일관되게 우수한 성능을 확보하기 어렵다는 점을 보여주며, 데이터 크기와 구조에 따라 적합한 MinPts가 결정되어야 함을 시사한다. 연산 시간 측면에서는 모든 문제 크기에서 MinPts=2일 때 가장 높은 연산 시간을 보였으나, MinPts 값이 증가함에 따라 연산 시간이 일관되게 감소하는 패턴은 나타나지 않았다.

### 5.3 방법론과 전략의 도입 효과

본 절에서는 군집화 도입 여부와 차고지의 유연한 할당을 고려하는 전략 도입 여부를 각각 비교하여 본 연구에서 제시하는 방법론과 전략의 전반적인 효과를 검증하고자 한다. 이

를 위해 각 인스턴스에서의 10회 반복 실험의 평균값에 대한 전체 평균값을 기준으로 비교분석을 진행하였다.

#### (1) 군집화 도입 효과

실험 결과, <Figure 8>은 군집화 도입 여부에 따른 평균 총 비용 차이를 나타낸다. 먼저 <Figure 8>의 첫 번째 그래프를 통해, 경로 개선 및 군집 할당 조정 알고리즘을 적용하기 전의 초기 비용은 군집화를 도입했을 때 더 낮음을 확인할 수 있다. 따라서, 군집화의 도입이 더 우수한 품질의 초기해를 형성하는데 기여함을 시사한다. 다음으로 두 번째 그래프에서도 경로 개선이 완료된 최종 비용 측면에서도 군집화를 도입한 방식이 유의미하게 더 낮은 값을 달성하는 것을 확인할 수 있다. 마지막으로, 세 번째 그래프를 살펴보면 문제 크기가 작을 때는 군집화 도입 여부가 연산 시간에 큰 영향을 주지 않지만, 수요지 수가 1,600, 1,800인 대규모 문제의 경우에는 군집화 도입 시 연산 속도가 유의미하게 개선됨을 알 수 있다. 따라서 군집화는 단순히 탐색 속도를 높이는 보조적 절차가 아니라, 탐색 시작점 자체를 구조화하여 전반적인 해 품질 개선에 기여하는 전략임을 확인하였다. 종합하면, 군집화는 대규모 문제 상황에서 해의 품질과 연산 효율성을 동시에 향상하는 핵심적 기법으로 기능한다. 특히 본 실험에서 확인된 바와 같이, 초기 해 단계부터 최종 해에 이르기까지 비용 절감 효과와 계산 시간 단축 효과가 일관되게 관찰되었다는 점에서, 군집화는 실용적 측면에서 매우 유의미한 접근임을 확인할 수 있다.

Table 4. Testing results of large-sized instances

Ins.	D	MinPts	Clusters	Average Cost(₩)		CPU(s)
				Initial Cost	Final Cost	
1	1,200	2	155	115,842	91,053	316
2		3	77	110,240	95,034	195
3		4	62	104,186	86,906	229
4		5	55	106,934	88,394	37
5	1,400	2	176	114,508	94,582	322
6		3	94	115,680	92,160	151
7		4	72	113,259	103,055	214
8		5	65	114,330	90,634	164
9	1,600	2	197	116,036	103,740	302
10		3	112	106,158	96,252	318
11		4	74	116,304	97,880	193
12		5	72	120,538	92,860	156
13	1,800	2	223	115,655	104,615	279
14		3	112	108,417	96,252	272
15		4	85	113,730	91,695	210
16		5	74	110,199	93,757	144
avg.				112,626	94,929	219

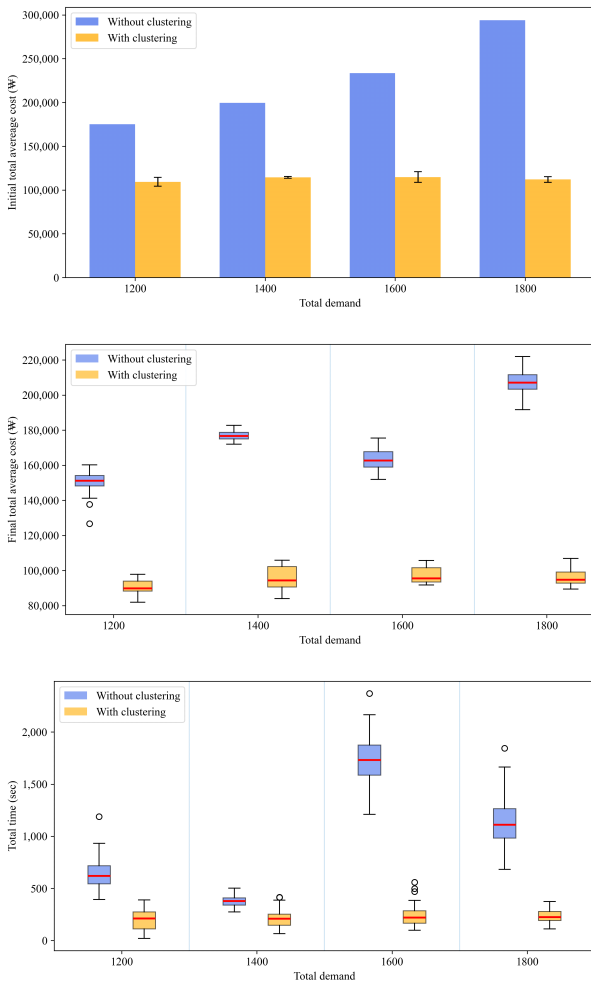


Figure 8. A Comparison of average costs and time with and without clustering

(2) 경로 계획 알고리즘의 효과

4.3절에서 설명한 경로 계획 알고리즘은 총 두 단계의 개선 과정을 통해 해를 개선한다. 초기 해 생성 단계에서는 최근접 이웃 탐색법을 통해 초기 해를 생성한다. 이후 첫 번째 개선 단계에서는 SA 알고리즘을 통해 일차적으로 해를 개선한다. 두

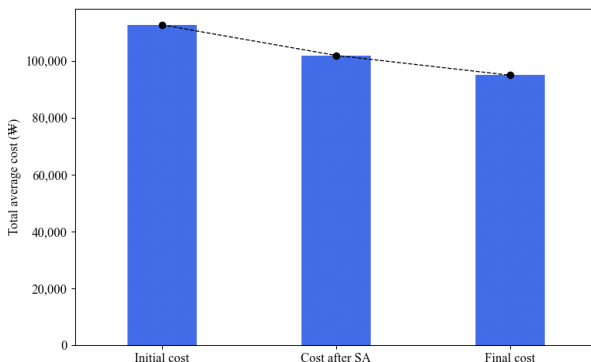


Figure 9. An improvement of total average cost

번째 개선 단계에서는 군집 할당 조정 알고리즘을 통해 군집을 이동시키며 최종적으로 해를 개선한다. <Figure 9>에 나타나 있는 것처럼, 초기 해 비용은 112,626원이고 첫 번째 단계에서의 비용은 101,884원으로 약 9.5% 개선되며, 마지막 단계에서의 비용은 94,929원으로 첫 번째 개선 단계 대비 약 6.8% 추가 개선된다. 즉, 본 연구에서 제안하는 경로 계획 알고리즘을 통해 단순 최근접 이웃 탐색법 대비 총 17,697원의 비용을 절감할 수 있다.

(3) 차고지의 유연한 할당을 고려하는 전략 도입 효과

실험 결과, <Figure 10>은 차고지의 유연한 할당을 고려하는 전략의 도입 여부에 따른 총 평균 비용 차이를 나타낸다. 구체적으로, 전략을 도입하지 않은 경우 차량 이동 비용 327,336원과 지연 비용은 69,319원이 발생하여 총 396,655원의 비용이 발생하였다. 반면, 전략을 도입한 경우 차량 운행 비용은 251,599원으로 약 23% 감소하였으며, 지연 비용은 7,908원으로 약 89% 절감되는 결과가 나타났다. 다만 7,615원의 차고지 이용 비용이 추가로 발생하였으나, 이동 및 지연 비용 절감 효과가 이를 상회하여 전체 비용은 267,122원으로 약 33% 감소하였다.

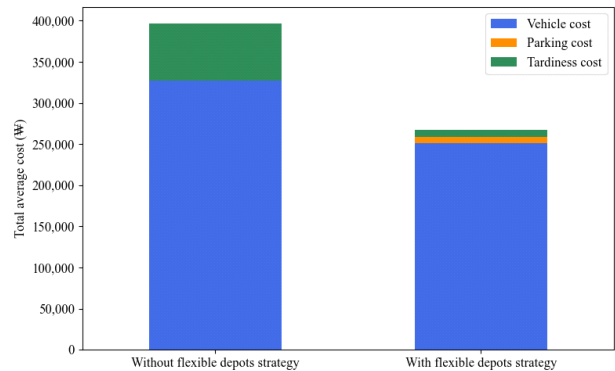


Figure 10. A comparison of total average costs with and without flexible depots strategy

이러한 결과는 차량이 출발 차고지로 반드시 복귀할 필요 없이 인근의 전기차 주차장을 차고지로 활용할 수 있게 되면 차고지의 유연한 할당을 고려하는 전략이 단순히 차량의 불필요한 이동을 줄이는 수준을 넘어, 지연 발생을 근본적으로 완화하는 효과를 제공함을 시사한다. 이는 도시 전역에 분산된 주차 자원을 활용함으로써, 물류 네트워크의 운영 안정성과 서비스 수준을 동시에 향상할 수 있음을 시사한다.

6. 결론

본 연구는 인천광역시의 ‘소상공인 택배 지원사업’을 사례로 차고지의 유연한 할당을 고려한 대규모 택배 수거 경로 계획

문제를 다루었다. 기존 단일 차고지 기반 운영 체계의 비효율성을 개선하기 위해 도시의 전기차 주차장을 추가적인 거점 차고지로 활용하는 전략을 제시하였으며, 해당 전략의 도입은 기존 단일 차고지 상황 대비 약 33%의 비용 절감 효과를 입증했다. 이는 영업일 250일 기준으로 환산 시 연간 약 32,000,000원의 비용 절감 효과로 이어진다. 이 수치는 본 연구의 목적함수만을 반영한 것으로, 실제 운영비 전반을 고려할 경우 비용 절감 효과는 더욱 확대될 수 있다. 또한, 대규모 문제의 계산 복잡도를 완화하기 위해 효율적으로 양질의 해를 탐색 가능한 군집화-차량 할당-경로 계획 3단계 접근법을 제시하였다.

본 연구는 몇 가지 한계를 가진다. 첫째, 실험이 인천광역시라는 특정 도시를 대상으로 수행되었으므로, 다른 지역이나 물류 네트워크 구조에서 동일한 효과가 나타난다고 단정하기 어렵다. 따라서 다양한 도시 사례를 통한 추가 검증이 필요하다. 둘째, 제안한 경로 계획은 SA에 기반하여 구현되었으나, 다른 메타휴리스틱 알고리즘이나 하이브리드 접근법과의 성능 비교는 이루어지지 않았다. 향후 연구에서는 본 연구의 접근법이 일반적이고 확장 가능한 도시 물류 전략으로 발전하도록, 다양한 물류 환경과 최적화 기법을 병행 검토하여 알고리즘의 상대적 우수성 및 활용 가능성을 명확히 규명하는 작업이 이루어져야 한다.

## 참고문헌

- Baals, J., Emde, S., and Turkensteen, M. (2023), Minimizing earliness-tardiness costs in supplier networks: A just-in-time truck routing problem, *European Journal of Operational Research*, **306**, 707-741.
- Ben-Ameur, W. (2004), Computing the initial temperature of simulated annealing, *Computational Optimization and Applications*, **29**, 369-385.
- Beyer, K. S., Goldstein, J., Ramakrishnan, R., and Shaft, U. (1999), When is "nearest neighbor" meaningful?, *Proceeding 7th International Conferences on Database Theory (ICDT '99)*, Jerusalem, Israel, Springer, 217-235.
- Crainic, T.G., and Toulouse, M. (2005), Parallel metaheuristics applications in transportation and logistics, in: Alba, E. (ed.) *Parallel Metaheuristics: A New Class of Algorithms*, Wiley-Interscience, Hoboken, NJ, 497-552.
- Ester, M., Kriegel, H. P., Sander, J., and Xu, X. (1996), A density-based algorithm for discovering clusters in large spatial databases with noise, *Proceeding 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96)*, Portland, OR, AAAI Press, 226-231.
- Fisher, M. L. and Jaikumar, R. (1981), A generalized assignment heuristic for vehicle routing, *Networks*, **11**(2), 109-124.
- Gillett, B. E. and Miller, L. R. (1974), A heuristic algorithm for the vehicle-dispatch problem, *Operations Research*, **22**(2), 340-349.
- Gu, Z., Zhu, Y., Wang, Y., Du, X., Guizani, M., and Tian, Z. (2022), Applying artificial bee colony algorithm to the multidepot vehicle routing problem, *Software: Practice and Experience*, **52**(3), 756-771.
- Hyundai Motor Company (2025), Porter II Electric, Hyundai Motor Company, Korea, <https://www.hyundai.com/kr/ko/e/vehicles/porter-2-electric/intro>.
- Incheon Facilities Corporation (2025), Parking fee guide, Incheon Facilities Corporation, <https://www.insiseoul.or.kr/life/parking/guide/fee.jsp>.
- Incheon Metropolitan City (2024a), Incheon Metropolitan City announces the "Small Business Half-Price Delivery" support project, Incheon Metropolitan City, Korea.
- Incheon Metropolitan City (2024b), "Incheon Small Business Half-Price Delivery" support project media briefing material (for distribution), Incheon Metropolitan City, Korea.
- Kek, A. G., Cheu, R. L., and Meng, Q. (2008), Distance-constrained capacitated vehicle routing problems with flexible assignment of start and end depots, *Mathematical and Computer Modelling*, **47**(1-2), 140-152.
- Kirkpatrick, S., Gelatt Jr, C. D., and Vecchi, M. P. (1983), Optimization by simulated annealing, *Science*, **220**(4598), 671-680.
- Korea Energy Agency (2025), Electric vehicle charging fees, Zero Emission Vehicle Integrated Portal, Korea, <https://ev.or.kr/nportal/evcarInfo/initEvcarChargePrice.do>
- Kraiem, A., Audy, J. F., and Lamghari, A. (2025), Mixed integer linear programming model for a multi-depot arc routing problem with different arc types and flexible assignment of end depot, *Transportation Research Procedia*, **82**, 1109-1119.
- Li, J., Dai, B.T., Niu, Y., Xiao, J., and Wu, Y. (2024), Multi-Type Attention for solving multi-depot vehicle routing problems, *IEEE Transactions on Intelligent Transportation Systems*, **25**(11), 17831-17840.
- Li, J., Li, Y., and Pardalos, P. M. (2016), Multi-depot vehicle routing problem with time windows under shared depot resources, *Journal of Combinatorial Optimization*, **31**, 515-532.
- Li, X., He, B., Zhao, Y., and Dong, Y. (2024), A two-stage optimization approach for multi-depot vehicle routing problem with time windows, *Proceeding 4th International Conference on Computer Communication and Artificial Intelligence (CCAI 2024)*, Xi'an, China, IEEE, 540-545.
- Mao, Z., Huang, D., Fang, K., Wang, C., and Lu, D. (2020), Milk-run routing problem with progress-lane in the collection of automobile parts, *Annals of Operations Research*, **291**, 657-684.
- Markov, I., Varone, S., and Bierlaire, M. (2016), Integrating a heterogeneous fixed fleet and a flexible assignment of destination depots in the waste collection VRP with intermediate facilities, *Transportation Research Part B: Methodological*, **84**, 256-273.
- Pu, X., Lu, X., and Han, G. (2022), An improved optimization algorithm for a multi-depot vehicle routing problem considering carbon emissions, *Environmental Science and Pollution Research*, **29**, 54940-54955.
- Robichet, A., Nierat, P., and Combes, F. (2022), First and last miles by cargo bikes: Ecological commitment or economically feasible? The case of a parcel service company in Paris, *Transportation Research Record*, **2676**(9), 269-278.
- Satopää, V., Albrecht, J., Irwin, D., and Raghavan, B. (2011), Finding a "Kneedle" in a haystack: Detecting knee points in system behavior, *Proceeding 31st International Conference on Distributed Computing Systems Workshops (ICDCSW)*, IEEE, 166-171.
- Schubert, E., Sander, J., Ester, M., Kriegel, H. P., and Xu, X. (2017), DBSCAN revisited, revisited: Why and how you should (still) use DBSCAN, *ACM Transactions on Database Systems*, **42**(3), Article

- 19.
- Soares, V. C. and Roboredo, M. (2024), On the exact solution of the multi-depot open vehicle routing problem, *Optimization Letters*, **18**(4), 1053-1069.
- Statistics Korea (2025), Average traffic speed, e-Nara Indicators, Korea, <https://www.index.go.kr/unify/idx-info.do?idxCd=5079>.
- Tillman, F. A. (1969), The multiple terminal delivery problem with probabilistic demands, *Transportation Science*, **3**(3), 192-204.
- Zhang, Z., Tan, S., Qin, J., Zou, K., and Zhou, S. (2025). Multi-strategy ant colony optimization with k-means clustering algorithm for capacitated vehicle routing problem, *Cluster Computing*, **28**(3), 202.

<부록 A>

DBSCAN의 선정 타당성을 검증하기 위해 군집 수를 자동으로 결정하는 대표적인 알고리즘인 Mean Shift, Affinity Propagation과의 비교 실험을 수행하였다. K-means와 같이 군집 수를 사전에 지정해야 하는 알고리즘은 매 사이클마다 수요 발생 위치와 물량이 달라지는 본 문제의 특성상 적정 군집 수를 사전에 결정하기 어렵고, 사이클 간격이 짧아 이를 매번 탐색하는 것은 운영상 현실적이지 않으므로 비교 대상에서 제외하였다. 각 알고리즘의 파라미터는 <Table A1>과 같이 설정하였으며, 문제 크기별로 10회 반복 실험을 수행하여 평균과 표준편차를 산출하였다.

Table A1. Parameter settings for clustering algorithms

Algorithm	Parameter Settings
DBSCAN	$\epsilon$ : auto (Kneedle method), MinPts = 3
Mean Shift	bandwidth: auto (estimate_bandwidth)
Affinity Propagation	preference = None, damping = 0.5

군집화 품질 지표로는 군집 내 최근접 이웃 이동 거리의 합 (Total Intra-cluster KNN Distance)을 사용하였다. 이는 본 연구의 경로 계획 단계에서 SA의 초기해를 최근접 이웃 탐색법으로 생성하는 점을 고려할 때, 경로 계획을 위한 군집화 품질 평가에 적합한 지표이다. 비교 실험 결과는 <Table A2>와 같다.

Table A2. Comparison results of clustering algorithms

D	Algorithm	Total Intra-Distance	
		Mean	Std.
1,200	DBSCAN	330.8	18.3
	Mean Shift	464.9	21.6
	Affinity Propagation	464.9	19.0
1,400	DBSCAN	351.8	21.9
	Mean Shift	496.8	32.6
	Affinity Propagation	482.7	27.4
1,600	DBSCAN	373.4	23.7
	Mean Shift	517.7	15.6
	Affinity Propagation	506.9	33.5
1,800	DBSCAN	378.9	21.2
	Mean Shift	534.9	15.8
	Affinity Propagation	518.6	34.7

실험 결과, DBSCAN이 모든 문제 크기에서 가장 낮은 군집 내 이동 거리를 기록하였다. 또한 표준편차 측면에서 DBSCAN은 18.3~23.7 범위로 일관된 값을 보인 반면, Mean Shift는 15.6~32.6, Affinity Propagation은 19.0~34.7로 문제 크기에 따른 변동 폭이 크게 나타났다. 이러한 결과는 DBSCAN이 낮은 이동 거리와 일관된 분산을 통해 강건한 군집화 품질

을 제공함을 보여준다.

<부록 B>

수거지의 위치는 공공데이터 포털의 인천광역시 지역구별 통신판매업 주소를 통합하여 활용하였다.

Table B1. Data sources by district

Region	Source
Don-gu	<a href="https://www.data.go.kr/data/15008536/fileData.do">https://www.data.go.kr/data/15008536/fileData.do</a>
Michuhol-gu	<a href="https://www.data.go.kr/data/15016234/fileData.do">https://www.data.go.kr/data/15016234/fileData.do</a>
Yeonsu-gu	<a href="https://www.data.go.kr/data/15039735/fileData.do">https://www.data.go.kr/data/15039735/fileData.do</a>
Nandong-gu	<a href="https://www.data.go.kr/data/15067119/fileData.do">https://www.data.go.kr/data/15067119/fileData.do</a>
Bupyeong-gu	<a href="https://www.data.go.kr/data/15148157/fileData.do">https://www.data.go.kr/data/15148157/fileData.do</a>
Gyeyang-gu	<a href="https://www.data.go.kr/data/15039184/fileData.do">https://www.data.go.kr/data/15039184/fileData.do</a>
Seo-gu	<a href="https://www.data.go.kr/data/15039509/fileData.do">https://www.data.go.kr/data/15039509/fileData.do</a>
Jung-gu	<a href="https://www.data.go.kr/data/3079659/fileData.do">https://www.data.go.kr/data/3079659/fileData.do</a>

<부록 C>

SA의 파라미터인 초기 열등해 수용 확률  $p_0$ 와 감가율  $\alpha$ 에 대한 실험을 수행하였다.  $p_0$  값의 범위는 적절한 초기 해 수용을 보장하기 위해  $p_0 = \{0.5, 0.7, 0.9\}$  범위를 고려하였고,  $\alpha$  값의 범위는 충분한 탐색을 확보하기 위해  $\alpha = \{0.950, 0.965, 0.980, 0.995, 0.999\}$  범위를 고려하였다. 실험은 <Table 4>에서 고려한 모든 인스턴스에 대해 파라미터 조합별 10회 반복 실험을 통한 평균값을 도출하였으며, 군집 할당 조정 알고리즘은 적용하지 않고 SA를 통한 경로계획 단계까지만 수행하였다.

먼저,  $p_0$ 의 효과를 평가하기 위해, 각 인스턴스별 모든  $\alpha$  값에 대한 평균을 비교하였다. <Table C1>는 이러한 방식으로 산출한 인스턴스 별 평균 비용을 보여주며, 각 인스턴스 별로 가장 작은 값을 보인  $p_0$ 를 나타낸다. 실험 결과 대전체 16개의 인스턴스 중 11개의 인스턴스에서  $p_0 = 0.5$ 에서 가장 준수한 성능을 보여 해당 값을 초기 해 수용 확률로 설정하였다. 다음으로, <Table C2>는  $p_0 = 0.5$ 일 때, 각  $\alpha$  별 비용과 연산 시간을 보여준다.  $\alpha$  값이 커짐에 따라 비용이 감소하고 연산 시간은 증가하는 경향을 보였다. 특히,  $\alpha = 0.999$ 는  $\alpha = 0.995$ 에 비해 추가로 평균적으로 약 8% 비용 개선을 이뤄내며, 시간은 평균적으로 약 5배 증가하였다. 본 연구에서는 군집 할당 조정 알고리즘 적용시 SA를 통한 지속적인 군집 할당을 평가하기 때문에, 추가적인 비용 개선 대비 연산 시간 증가가 급격해지는 구간을 회피하면서도 충분한 비용 개선을 달성할 수 있는  $\alpha = 0.995$ 를 감가율로 채택하였다.

**Table C1.** Average cost by  $p_0$

Ins.	$p_0$	Average Cost(₩)	Ins.	$p_0$	Average Cost(₩)	Ins.	$p_0$	Average Cost(₩)	Ins.	$p_0$	Average Cost(₩)
1	0.5	107,609	5	0.5	103,076	9	0.5	108,072	13	0.5	108,726
	0.7	107,808		0.7	103,047		0.7	107,888		0.7	108,636
	0.9	107,777		0.9	103,024		0.9	108,395		0.9	108,661
2	0.5	100,794	6	0.5	109,400	10	0.5	99,785	14	0.5	101,729
	0.7	100,830		0.7	109,401		0.7	99,796		0.7	101,594
	0.9	100,830		0.9	109,439		0.9	99,972		0.9	101,757
3	0.5	93,667	7	0.5	107,383	11	0.5	107,325	15	0.5	105,579
	0.7	93,985		0.7	107,559		0.7	107,862		0.7	105,742
	0.9	94,176		0.9	107,667		0.9	107,568		0.9	105,830
4	0.5	93,850	8	0.5	103,421	12	0.5	115,914	16	0.5	102,707
	0.7	93,764		0.7	103,438		0.7	116,069		0.7	102,780
	0.9	93,914		0.9	103,562		0.9	116,134		0.9	102,779

**Table C2.** Average cost and CPU time by  $\alpha$

Ins.	$\alpha$	Average Cost(₩)	CPU (s)	Ins.	$\alpha$	Average Cost(₩)	CPU (s)	Ins.	$\alpha$	Average Cost(₩)	CPU (s)	Ins.	$\alpha$	Average Cost(₩)	CPU (s)
1	0.950	111,070	13	1	0.950	110,427	13	1	0.950	112,589	14	1	0.950	113,890	17
	0.965	110,806	16		0.965	109,692	16		0.965	112,417	18		0.965	113,916	21
	0.980	110,394	24		0.980	107,264	24		0.980	112,073	27		0.980	113,770	31
	0.995	105,724	71		0.995	97,264	78		0.995	106,944	86		0.995	107,287	99
	0.999	100,052	324		0.999	90,733	361		0.999	96,337	399		0.999	94,769	454
2	0.950	103,881	13	2	0.950	112,605	14	2	0.950	104,024	14	2	0.950	106,234	17
	0.965	103,304	16		0.965	112,391	17		0.965	103,869	18		0.965	106,471	21
	0.980	103,398	22		0.980	112,108	25		0.980	103,449	26		0.980	106,312	31
	0.995	98,376	70		0.995	107,695	80		0.995	98,184	86		0.995	99,694	96
	0.999	95,009	317		0.999	102,200	364		0.999	89,399	402		0.999	89,936	443
3	0.950	99,097	12	3	0.950	109,581	14	3	0.950	113,609	14	3	0.950	111,585	17
	0.965	98,583	15		0.965	109,348	17		0.965	113,512	17		0.965	110,998	21
	0.980	97,965	22		0.980	108,873	25		0.980	112,826	26		0.980	109,632	31
	0.995	89,720	69		0.995	106,962	78		0.995	102,853	85		0.995	103,047	96
	0.999	82,972	313		0.999	102,150	360		0.999	93,824	398		0.999	92,635	442
4	0.950	100,542	14	4	0.950	108,830	14	4	0.950	118,802	13	4	0.950	107,607	16
	0.965	100,035	16		0.965	108,539	17		0.965	118,674	16		0.965	107,449	20
	0.980	98,248	23		0.980	108,097	25		0.980	118,364	24		0.980	106,772	29
	0.995	88,389	69		0.995	102,250	79		0.995	115,051	84		0.995	100,713	95
	0.999	81,605	310		0.999	89,387	358		0.999	108,679	399		0.999	90,992	442

**저자소개**

**김현준:** 인천대학교 산업경영공학과에서 2025년 학사학위를 취득하고 인천대학교 산업경영공학과 석사과정에 재학 중이다. 주요 연구분야는 경제성공학, 수리경제학, 강화학습이다.

**김예찬:** 인천대학교 산업경영공학과에서 2025년 학사학위를 취득하고 인천대학교 산업경영공학과 석사과정에 재학 중이다.

주요 연구분야는 비즈니스 애널리틱스, 데이터 마이닝이다

**김병수:** 2009년 미국 Auburn대학교 산업및시스템공학과에서 박사학위를 취득하였다. 2013년부터 인천대학교 산업경영공학과에서 교수로 재직하고 있다. 주요 연구분야는 SCM, 생산/물류 최적 운영 계획 및 통제이며 관심해법은 수리계획법, 메타휴리스틱 알고리즘, 머신러닝, 강화학습 등이다.