Optimizing Urban Logistics: Vehicle Routing Problem With Underground Transportation

Ho Young Jeong and Byung Duk Song

***Abstract*— In response to increasing congestion and inefficiencies in urban freight transportation systems, driven by the rapid rise of e-commerce and urbanization, underground logistics are emerging as a promising alternative. This study introduces the Vehicle Routing Problem with Underground Transportation (VRP-UT), a novel model that integrates traditional vehicle routing with underground logistics facilities to optimize urban deliveries. To address the model's complexity, we propose a hybrid solution approach that combines Q-learning with pruning techniques, enhancing route optimization and minimizing unnecessary operations. Comprehensive numerical and sensitivity analyses further validate the model's effectiveness, positioning VRP-UT as a scalable and efficient solution to meet growing urban logistics demands while supporting sustainable development. A case study, including sensitivity analysis based on Seoul Subway Line 3, demonstrates the practical applicability of the VRP-UT model, highlighting its potential to alleviate surface-level traffic congestion, reduce operational costs, and shorten delivery times in densely populated urban areas.**

***Index Terms*— Urban logistics, urban congestion management, last-mile delivery, vehicle routing, mathematical model, metaheuristic.**

I. Introduction

The exponential growth in e-commerce and express delivery sectors has led to a remarkable increase in logistics demand, significantly altering market dynamics. As reported by the United Nations Conference on Trade and Development (UNCTAD), the share of e-commerce in retail sales jumped from 16% to 19% in 2020 [58]. Similarly, forecasts by the Organization for Economic Co-operation and Development (OECD) suggest that global surface freight ton-kilometers may double by 2030 from their 2015 figures, with a notable 3.2-fold increase expected in Asian regions [50]. This rapid growth, while crucial to urban economic advancement, also poses considerable challenges for urban distribution networks [53]. Increasing vehicle numbers and worsening traffic congestion further strain existing urban ground road infrastructures, thus hindering logistics and distribution efficiency [9]. In this context, developing strategies for timely and efficient delivery becomes a crucial research area in urban distribution.

Traditional freight movement methods face significant challenges within densely populated urban areas [48]. Various strategies have been proposed to ease urban traffic congestion. These include expanding traditional road network capacities, often limited by the tight urban spaces of modern cities. The rise of shared mobility, expected to grow substantially, presents a possible solution to urban transport issues [49]. Recent studies have also introduced multi-objective formulations for electric vehicle routing problems, including shuttle bus fleets at university campuses [25] and unmanned vehicle fleets in large-scale urban logistics systems [71], offering valuable advancements for optimizing urban logistics. However, this approach primarily serves individual transportation needs, making it more suitable for residents and less comprehensive for broader urban transportation dilemmas [45]. Air delivery systems, especially drones, have also been recognized as a promising option for diverse logistical challenges [28], [29], [30]. Nevertheless, their adoption is hindered by air traffic safety and data privacy concerns [11], [46].

Against this backdrop, Underground Logistics Systems (ULS) have emerged as an innovative solution, offering numerous benefits for urban distribution. The ULS concept involves using subterranean channels for logistics operations to alleviate surface traffic congestion, reduce operational costs, and improve delivery efficiency [60]. This novel approach, which includes using underground rail resources, represents a significant shift in urban logistics, paving the way for more sustainable and efficient urban development. ULS also aligns with sustainable urban development goals, introducing a transformative logistics model with modular autonomous vehicles and dynamic pickup and delivery systems [10], [52].

Ongoing research into ULS for nearly 3 decades has consistently highlighted its considerable advantages. ULS systems, requiring minimal surface space, offer resilience against extreme weather impacts on logistics, enhancing the reliability and safety of delivery services. Furthermore, using ULS as a supplementary freight delivery method can significantly ease surface transportation burdens, reducing highway maintenance expenses and decreasing truck-related accidents.

ULS stands out as a sustainable option in urban logistics, deserving serious consideration as a practical transportation mode. Globally, the development of ULS has attracted significant academic interest, with many countries increasingly exploring the use of underground spaces. Like an urban

Received 16 May 2024; revised 5 November 2024 and 11 January 2025; accepted 21 January 2025. Date of publication 4 February 2025; date of current version 5 May 2025. This work was supported in part by BK21 Fostering Outstanding Universities for Research (FOUR) funded by the Ministry of Education of Korea and the National Research Foundation of Korea (NRF), in part by the Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korea Government [Ministry of Science and ICT (MSIT)] under Grant RS-2024-00438411, and in part by the NRF grant funded by the Korea Government (MSIT) under Grant RS-2022-NR070854. The Associate Editor for this article was P. Wang. *(Corresponding author: Byung Duk Song.)*

Ho Young Jeong is with the Research Institute of Engineering and Technology, Korea University, Seoul 02841, South Korea (e-mail: ghy27@korea.ac.kr).

Byung Duk Song is with the School of Industrial and Management Engineering, Korea University, Seoul 02841, South Korea (e-mail: bdsong@korea.ac.kr).

Digital Object Identifier 10.1109/TITS.2025.3533477

subway system, ULS uses subterranean pipes or tunnels to transport goods, reducing the load on surface roads and lessening urban traffic congestion, as shown in Figure 1. Additionally, using clean energy in these systems further lowers urban pollution, while their immunity to external environmental factors enhances operational efficiency and reliability. The resulting decrease in ground truck traffic also leads to considerable environmental and economic benefits, such as lower road maintenance costs, helping to offset the substantial initial investment needed for ULS infrastructure [68], [73].

Our work advances urban logistics optimization by integrating reinforcement learning with exact methods, specifically designed for ULS. While progress has been made in ULS network design, most research focuses on isolated technological aspects and does not address the full integration with surface-level transport systems. Furthermore, many existing models rely on traditional optimization methods, which are often too rigid to accommodate the dynamic nature of urban logistics, especially when managing the complexities of both underground and surface transportation networks. Our study addresses these gaps by introducing novel solutions that seamlessly combine underground and surface transport systems, enabling dynamic and adaptive route optimization. The key contributions of our research are as follows:

1) **Introduction of the Vehicle Routing Problem with Underground Transportation (VRP-UT):** We introduce the VRP-UT model, integrating underground stations into the conventional vehicle routing framework. This innovative model allows for more efficient urban deliveries by leveraging underground transport to alleviate surface-level congestion, leading to optimized route selection for last-mile logistics.
2) **Hybrid Q-learning with Pruning for Dynamic Route Optimization:** Our hybrid framework, combining Q-learning with pruning techniques, dynamically adjusts to changing conditions such as traffic, delays, and operational disruptions. This enhanced adaptability improves route optimization and increases the efficiency of urban logistics operations in complex, rapidly changing environments.
3) **Numerical and Sensitivity Analyses with Case Study:** While many studies have relied on theoretical models or limited case studies, we validate our approach through comprehensive numerical analyses and a case study based on Seoul Subway Line 3. This case study demonstrates the practical benefits of our approach, showing significant reductions in operational costs and delivery times, while offering actionable insights for guiding future urban logistics implementations in other metropolitan areas.

These contributions address the limitations of existing research by combining underground systems with surface transport and introducing a flexible optimization framework. Our work bridges traditional VRP methods with advanced machine learning techniques, providing a scalable and adaptable solution for future urban transportation systems.

II. Literature Review

Urban development and the continuous expansion of metropolitan areas have brought about challenges such as traffic congestion, pollution, and space constraints, which not only hinder the efficiency of logistics but also have significant negative impacts on the environment and urban mobility [6], [22]. Urban logistics is crucial for ensuring the timely and cost-effective delivery of goods, yet conventional surface-level transportation methods face limitations in addressing these challenges [32]. Moreover, the integration of electric vehicles in urban logistics systems is an emerging solution, providing a more sustainable and efficient way to navigate city infrastructures [26]. As cities grow in population and economic activity, the strain on existing logistics infrastructure intensifies, resulting in inefficiencies and increased operational costs [24], [51].

The integration of underground systems within urban logistics networks offers a promising solution and has increasingly been considered as a viable option to address urban freight movement by making use of the subterranean spaces available in dense urban environments [16], [23]. Leveraging advanced infrastructure and technological solutions, ULS can provide an alternative to traditional road-based delivery, helping alleviate congestion, reduce pollution, and optimize the use of urban space [41], [69]. Early studies have focused on the feasibility and implementation of ULS, exploring the technical and economic aspects of integrating underground transport systems with urban logistics [3], [19]. While these studies have established the potential benefits of ULS, they often do not fully explore the integration of underground logistics with surface-level transportation networks, leaving a gap in addressing the broader operational and strategic challenges of urban logistics [21], [40].

A. Operational Optimization in Urban Logistics

Operational optimization is crucial for enhancing the efficiency and adaptability of urban logistics systems. Various studies have developed frameworks for improving routing, scheduling, and system flexibility, focusing on the integration of autonomous vehicles [34], multi-objective optimization methods such as truck-drone collaboration [42], and flexible routing systems [36]. These advancements help optimize transportation resources and reduce operational costs in urban logistics.

Beyond operational efficiency, environmental concerns are increasingly being integrated into the design and operation of ULS. Many urban logistics solutions now incorporate eco-friendly practices to minimize the environmental impact of freight movement. The adoption of sustainable energy solutions within ULS, such as electric vehicles and clean energy sources, is becoming more widespread. Furthermore, integrating underground logistics can significantly reduce surface-level congestion, indirectly lowering vehicle emissions and contributing to urban sustainability [24]. Additionally, green logistics routing approaches—such as eco-packaging and energy-efficient delivery routes [6]—are key components of ULS, aligning operational efficiency with environmental goals.

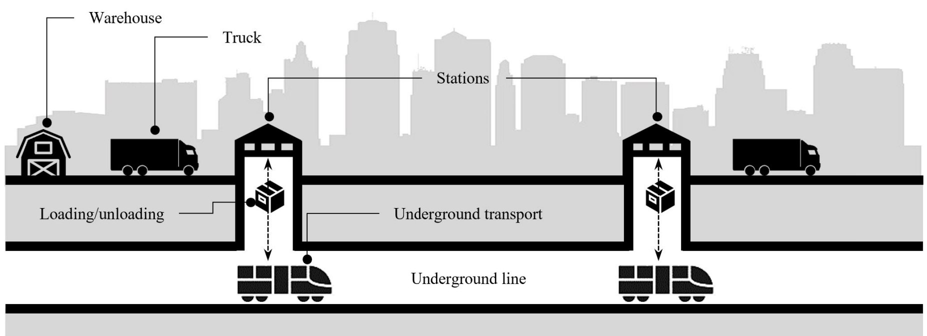

Fig. 1. Schematic representation of the integrated urban logistics system that combines surface truck routing with an underground logistics network.

Recent studies have examined how environmental considerations can be integrated into ULS frameworks, with particular emphasis on optimizing routing for food logistics systems, which enhances the sustainability and efficiency of urban transportation networks [39]. For instance, Schiewe and Schöbel [55] explores how underground logistics systems can reduce urban emissions while enhancing delivery efficiency. Similarly, Hulagu and Celikoglu [24] demonstrates that integrating underground logistics with traditional surface transport methods can alleviate congestion and lower carbon footprints. These findings underscore the importance of combining operational optimization with sustainable practices in ULS.

B. Multi-Modal Logistics Systems

The integration of various transport modes within ULS is crucial for improving urban logistics efficiency. As urban areas continue to face congestion and space constraints, combining underground and surface-level transport systems has become an important research area. Recent work emphasizes the integration of autonomous vehicles, drones, and traditional ground-based vehicles for enhanced multi-modal operations [42], as well as exploring optimization techniques for distributing goods within urban rail and underground systems [37]. Moreover, Zheng et al. [72] have examined urban rail service design for collaborative passenger and freight transport, optimizing logistics through the effective use of urban metro lines.

In addition to operational efficiency, sustainability is a key factor in multi-modal logistics systems. Recent studies have explored how ULS can reduce environmental impacts by integrating electric vehicles, green packaging, and sustainable energy sources into operations. Research by Hulagu and Celikoglu [24] highlights that integrating underground logistics not only alleviates surface-level congestion but also contributes to reducing carbon footprints. Moreover, eco-friendly routing algorithms [6] and the use of clean energy vehicles are key innovations driving sustainability within urban logistics systems. These advances are further supported by research, such as Schiewe and Schöbel [55], which investigates how environmental considerations can be integrated into ULS frameworks.

Recent research has also focused on multi-depot and resource-sharing strategies in urban logistics, which are essential for improving ULS network efficiency. Studies like Wang et al. [63], which explores resource sharing in multi-depot vehicle routing, and Wang et al. [61], which examines eco-packages, highlight the critical role of sustainability in enhancing logistics efficiency. Furthermore, Wang et al. [64] and Wang et al. [62] contribute to multi-depot logistics models, offering new strategies for better resource allocation and time-dependent logistics. These studies underscore the importance of combining multi-modal systems with sustainable practices to achieve both operational optimization and reduced environmental impact.

C. ULS Design and Optimization for Last-Mile Delivery

The design and optimization of ULS have been central to urban logistics research. Numerous studies have explored the integration of underground systems with urban freight networks, proposing various models and optimization techniques. Foundational work by Hulagu and Celikoglu [22], Henry et al. [19], and Binsbergen and Bovy [3] has provided valuable insights into the potential for integrating underground transportation systems within urban freight networks. These early studies addressed the technical feasibility, infrastructure requirements, and operational aspects of underground systems, setting the stage for further exploration of ULS.

As demand for efficient last-mile delivery solutions continues to rise, the design and optimization of ULS have become even more critical. Several studies have focused on innovations in underground transport systems and their integration with metro systems, electric vehicles, and other above-ground transportation methods. Research by Pei et al. [51], Ma et al. [43], and Cui and Nelson [8] laid the groundwork for integrating underground transport with urban freight networks. While significant contributions have been made, gaps remain in comprehensive approaches that effectively combine the operational strategies of underground logistics with surface-level transportation systems. Additionally, the practical challenges of integration—such as coordination, resource sharing, and operational efficiency—remain underexplored.

Recent advancements in ULS design and last-mile delivery solutions focus on optimizing distribution paths within urban rail systems and metro-based logistics frameworks. For instance, Leng and Li [37] and Hu et al. [21] explore the optimization of distribution paths for intelligent logistics vehicles within underground systems. Furthermore, studies by Wei et al. [66] and Mo et al. [47] propose novel last-mile delivery approaches, such as self-pickup modes and optimized routing within ULS frameworks. Hu et al. [20] also explore the use of ULS for automated waste collection and parcel delivery, offering new perspectives on the versatility of underground logistics in urban systems.

Despite the advancements in ULS network design and optimization, several limitations remain. Current models often focus on specific technological or operational aspects of ULS without fully integrating the complexities of multi-modal logistics systems. The integration of underground and surface-level transport systems remains underexplored, and many studies focus on isolated case studies or simulations, limiting the generalizability of the results. Additionally, existing research often relies on traditional optimization methods, which are not always adaptable to the dynamic nature of urban logistics.

This study addresses these limitations by presenting a comprehensive framework that integrates ULS with traditional ground-level delivery systems. The proposed model incorporates multi-depot arrangements, time windows, and inventory constraints, providing a holistic approach to urban logistics optimization. The study also introduces advanced metaheuristic techniques, such as reinforcement learning and exact optimization methods, which offer a dynamic, timely approach to solving urban logistics challenges. By integrating underground and surface-level logistics systems, this study provides a novel solution to urban logistics problems and sets new benchmarks for operational efficiency, sustainability, and flexibility in urban logistics systems.

III. Problem Description

This research introduces the VRP-UT, a novel model for last-mile delivery networks. The VRP-UT extends the classical multi-depot Vehicle Routing Problem (MDVRP) by incorporating inventory constraints and time windows, while also leveraging underground transfer stations to optimize delivery routes. The model is designed to facilitate not only deliveries but also pickups, drop-offs, and inventory transfers within a network of underground stations.

The VRP-UT logistics network consists of three main types of nodes: warehouses, stations, and customers. Warehouses function as inventory hubs where goods are stored and vehicles are loaded. Stations serve as transfer points within the underground system, facilitating the movement of goods between warehouses, especially when certain products are unavailable at the originating warehouse. Finally, customers are the ultimate delivery points, often requiring goods sourced from multiple warehouses or stations.

The model operates under the following key assumptions:

- Not all products are available at the originating warehouse, necessitating transfers between warehouses via underground stations.
- The underground transit system is used exclusively for transfers between stations, while vehicles handle the final deliveries to customers.
- Underground transfers are constrained by fixed schedules, requiring synchronization of vehicle routes with station times.
- Surface routes are bidirectional, allowing free movement between surface nodes, but underground transfers are restricted to scheduled times, adding a time-sensitive element to routing.
- When a transfer is required for serving a customer, the station must be visited to pick up the transferred item, and the station transferring the item must also be visited by the designated vehicle.

The operational dynamics of VRP-UT are illustrated in Figure 2, which contrasts traditional ground-level delivery with the integrated underground transfer system. n the traditional approach, illustrated by the example of vehicle routing problems with inventory constraints (VRP-IC) in Figure 2(a), warehouses directly service customers via ground routes. In contrast, the integrated VRP-UT approach (Figure 2(b)) uses underground stations as intermediate nodes for redistributing goods, reducing the need for long-distance ground travel. This ensures that goods are moved closer to their final destinations before the last leg of delivery.

By combining surface and underground logistics, the VRP-UT model offers a flexible and efficient solution for urban logistics challenges. The use of underground stations for inventory transfers reduces surface traffic, speeds up deliveries, and enhances overall logistics efficiency, particularly in congested city environments.

A. Notations

In this section, we delineate the notations utilized in formulating the VRP-UT. Consider the following sets and parameters:

Sets:

Symbol	Description
V	Set of vehicles.
W	Set of warehouse nodes.
I	Set of delivery nodes.
S	Set of station nodes.
N	Set of all nodes, $N = W \cup I \cup S$.
P	Fixed schedules of underground travel between stations, including departure and arrival times and corresponding stations.

Systemic Parameters:

Symbol	Description
T_{ij}	Travel time between nodes i and j, $((i, j) \in A)$.
v_T	Velocity for vehicle travel.
v_U	Velocity for underground travel.
W_v	Designated warehouse for vehicle v, $(v \in V)$.
W_i	Designated warehouse for delivery i, $(i \in I)$.
W_s	Designated warehouse for station s, $(s \in S)$.
W_i^{Inv}	Warehouse that has the inventory of the desired product for customer i, $(i \in I)$.
M	Large positive number.
Dpt^p	Departure time of schedule p, $(p \in P)$.
Arr^p	Arrival time of schedule p, $(p \in P)$.

Decision Variables:

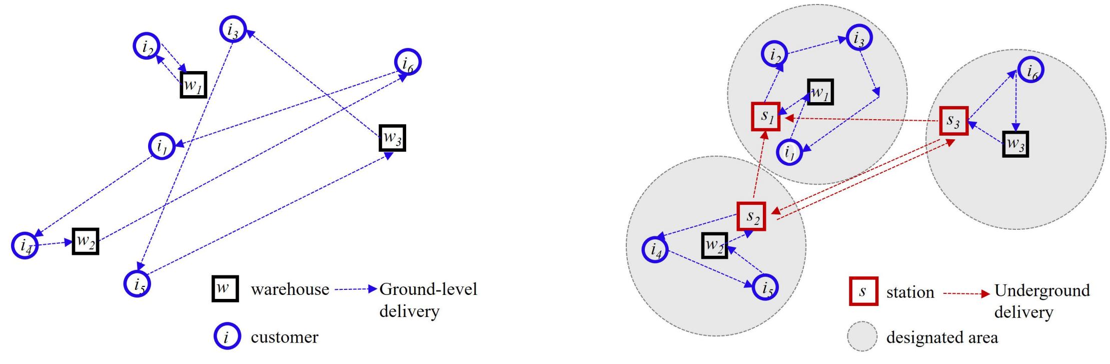

Fig. 2. An illustration of the delivery network: (a) traditional ground-level delivery (VRP-IC), (b) integrated ground-level and underground delivery (VRP-UT).

Symbol	Description.
x_{vij}	Binary variable; 1 if vehicle v travels from node i to node j, 0 otherwise.
t_{vi}^T	Travel time for vehicle v at node i.
$t^T a_v$	Arrival time for vehicle v after serving all customers.
$t_{\text{drop},s}^U$	Drop time at station s.
$t_{\text{pick},s}^U$	Pick-up time at station s.
$z_{s,s'}$	Binary variable for inventory transfer from station s to s'.
$U_{p,s,s'}$	Binary variable for selecting schedule p between stations s and s'.
$\mathcal{C}$	Completion time after all deliveries and returning.

The problem considers a set V representing the vehicles utilized for transportation. The set W denotes the warehouse nodes crucial to the distribution network. The set I corresponds to the customer nodes, which are the destinations for the goods transported. Additionally, the set S represents station nodes that play an integral role in the logistics process, particularly within the underground logistics segment.

Moreover, the set N encompasses all nodes in the network. It is defined as the union of warehouse, delivery, and station nodes, formally represented as $N = W \cup I \cup S$. The set A comprises all feasible arcs (i, j) in the network, where i and j are distinct nodes in N. For underground logistics, the set UG represents the arcs between underground stations.

Systemic parameters are pivotal in defining the dynamics of the problem. The parameter T_{ij} specifies the travel time between any two nodes i and j, assuming the shortest path. The cost parameters, C_T and C_U, represent the per-unit travel time costs for vehicles on the surface and underground, respectively. Additionally, W_v, W_i, and W_i^{Inv} denote the designated warehouses for vehicles, deliveries, and inventory respectively. The parameter $schedule_{sch}$ provides the available schedules for underground travel between stations.

Key decision variables include x_{vij}, a binary variable that is 1 if vehicle v travels from node i to node j, and 0 otherwise. The variables t_{vi}^T, $t_{\text{drop},s}^U$, and $t_{\text{pick},s}^U$ capture the travel time for vehicles at node i and the drop and pick times at station s, respectively. The variable $z_{s,s'}$ is a binary variable indicating whether an inventory transfer occurs from station s to s'. The variable $U_{sch,s,s'}$ represents the selection of a specific schedule sch for underground inventory transfers between stations s and s'. Lastly, $\mathcal{C}$) denotes the completion time after all deliveries are completed and returning to each depot.

B. Mixed-Integer Linear Programming Formulation

We now formulate the problem as a mixed-integer linear program (MILP), where the objective is to minimize the total completion time.

1) Objective Function: The primary goal of the model, as presented in Equation (1), is to minimize the total completion time. This objective focuses on optimizing vehicle routing in combination with underground logistics.This objective seeks to minimize the total completion time across all vehicles, denoted by $\mathcal{C}$.

$$\text{minimize}\, \mathcal{C} \tag{1}$$

2) Constraints:

a) Truck movement constraints: Constraint (2) ensures that each delivery is assigned to exactly one vehicle. This guarantees that every customer node is visited by one and only one vehicle. Constraint (3) restricts vehicles to only serve deliveries originating from their designated warehouses. This ensures that vehicles adhere to their respective warehouse assignments and do not deliver goods from other warehouses. Constraints (4) and (5) ensure that each vehicle starts and returns to its assigned warehouse. Vehicles must leave and return to their designated depot, ensuring efficient and logical routes. This restriction enforces the rule that vehicles are confined to their respective warehouse routes. Constraint (6) enforces flow balance at each node. For every vehicle, the number of arcs entering a node must equal the number of arcs leaving the node, ensuring proper flow continuity throughout the network.

$$\sum_{v \in V} \sum_{j \in N} x_{vji} = 1 \quad \forall i \in I \tag{2}$$

$$\sum_{v \in V} \sum_{j \in N} x_{vji} = 0 \quad \forall i \in I, v \neq w_i \tag{3}$$

$$\sum_{j \in N} x_{vwj} = 1 \quad \forall v \in V \tag{4}$$

$$\sum_{j \in N} x_{vjw} = 1 \quad \forall v \in V \tag{5}$$

$$\sum_{j \in N} x_{vji} = \sum_{j \in N} x_{vij} \quad \forall v \in V, i \in N \tag{6}$$

b) Time constraints for vehicles: Constraint (7) ensures that the travel time between nodes i and j is consistent with the actual travel duration. This guarantees that vehicles follow realistic time frames for moving between locations. Constraint (8) calculates the arrival time of each vehicle at its warehouse after completing all deliveries. This ensures that the vehicle routing concludes with the vehicle returning to its initial starting warehouse. Constraint (9) defines the total completion time $\mathcal{C}$ as the maximum of all vehicle arrival times, which establishes the overall time required to finish the delivery process for all vehicles.

$$t_{vj}^T \geq t_{vi}^T + T_{ij} - M(1 - x_{vij}) \quad \forall v \in V, (i, j) \in A \tag{7}$$

$$t_{a_v}^T \geq t_{vi}^T + T_{iw_v} - M(1 - x_{vij}) \quad \forall v \in V, i \in N \tag{8}$$

$$C_{\text{completion}} \geq t_{a_v}^T \quad \forall v \in V \tag{9}$$

c) Underground movement constraints: Constraint (10) ensures that if the inventory required for a delivery is located in a different warehouse, the vehicle must visit the corresponding drop station before making the delivery. Constraint (11) enforces that the vehicle also visits the pickup station associated with the delivery's designated warehouse. Constraint (12) guarantees that an underground transfer takes place between the drop s and pickup s' stations when there is a warehouse mismatch.

$$\sum_{v \in V} \sum_{j \in N} x_{v,j,s_{\text{drop}}} \geq 1 \quad \forall i \in I, s_{\text{drop}} = W_i^{\text{Inv}} \neq W_i \tag{10}$$

$$\sum_{v \in V} \sum_{j \in N} x_{v,j,s_{\text{pickup}}} \geq 1 \quad \forall i \in I, s_{\text{pickup}} = W_i \neq W_i^{\text{Inv}} \tag{11}$$

$$z_{s_{\text{drop}}, s_{\text{pickup}}} \geq 1 \quad \forall i \in I, W_i^{\text{Inventory}} = s_{\text{drop}}, W_i = s_{\text{pickup}} \tag{12}$$

d) Underground travel time constraints: Constraint (13) ensures $U_{p,s,s'}$ is set to 1 only if a valid schedule exists for a transfer between stations s and s', synchronizing the transfer with the schedule and $z_{s,s'}$. Building on this, Constraint (14) aligns the drop time at station s with the departure time of the schedule, avoiding delays. Similarly, Constraint (15) aligns the pick-up time at station s' with the schedule's arrival time, ensuring timely inventory transfers.

$$\sum_{p \in P} U_{p,s,s'} = z_{ss'} \quad \forall s, s' \in S \tag{13}$$

$$\text{Dpt}^p \geq t_{\text{drop},s}^U - M(1 - U_{sch,s,s'}) \quad \forall s \in S \tag{14}$$

$$t_{\text{pick},s'}^U \geq \text{Arr}^p - M(1 - U_{sch,s,s'}) \quad \forall s' \in S \tag{15}$$

e) Vehicle time constraints with underground transfers: To ensure seamless coordination between vehicle schedules and underground transfer operations, we introduce constraints that synchronize pick-up and drop-off times and address inventory mismatches. Constraint (16) ensures that when a vehicle v visits station s, its pick-up time t_{vs}^T aligns with the underground transfer's scheduled pick-up time $t_{\text{pick},s}^U$. Constraint (17) aligns the vehicle's drop-off time at station s with its travel time from node i, maintaining the correct sequence of movements.

$$t_{vs}^T \geq t_{\text{pick},s}^U - M(1 - x_{vis}) \quad \forall v \in V, s \in S \tag{16}$$

$$t_{\text{drop},s}^U \geq t_{vi}^T + T_{is} - M(1 - x_{vis}) \quad \forall v \in V, i \in N \tag{17}$$

The equations represent the core constraints of the model, ensuring effective vehicle routing, time management, and synchronization with the underground logistics network. These constraints are designed to optimize the paths and schedules, ultimately minimizing the total completion time $\mathcal{C}$.

IV. Q-Learning With Pruning

We propose Q-learning with Pruning (QLP) as a solution to the computational challenges of the VRP-UT, an NP-hard extension of the classical VRP. The transfer-based vehicle routing framework with time constraints presents immense complexity. Traditional reinforcement learning approaches encounter substantial difficulties with NP-hard problems due to the vast solution space and the challenge of locating optimal solutions efficiently [38]. QLP integrates pruning directly into the Q-learning process, dynamically reducing the solution space by eliminating suboptimal routes. This focus on promising paths mitigates computational limitations, enabling more efficient exploration within this domain.

The core idea behind QLP is to prune routes that violate specific optimality conditions, such as ensuring that the travel time before reaching a transfer station does not exceed the designated transfer time. Additionally, QLP respects preemption constraints, which require that specific nodes are visited in a predefined sequence following the station visit. By incorporating pruning, QLP dynamically removes unviable paths, allowing the Q-learning algorithm to more efficiently explore the solution space. The following sections outline how various methods, including Greedy algorithms, Minimum Spanning Tree limits, and pruning conditions, are applied within QLP to address the challenges of VRP-UT.

A. Greedy Algorithm: Upper Limit for Feasibility

In the context of QLP, the Greedy Algorithm provides an essential upper limit, serving as a baseline estimate of the total travel time. By generating a feasible solution with minimized immediate travel time, this cap allows QLP to identify and prune routes that become excessively costly relative to heuristic limits.

Theorem 1: Let C^G represent the total cost of the route generated by the Greedy Algorithm (see Appendix A), and let C^* denote the Hamiltonian cycle with the optimal cost for the routing problem. Then the following inequality holds:

$$C^G \geq C^*,$$

where C^G is an upper limit on the optimal solution C^* for the routing problem for each warehouse.

Proof: Consider the vehicle routing problem, where:

- V represents the set of nodes (deliveries and pickups) to be visited by the vehicle.

- $d(i, j)$ denotes the travel cost (e.g., distance or time) between nodes i and j, where $i, j \in V \cup \{0\}$, and 0 represents the depot.

The Greedy Algorithm constructs a feasible solution C^G by iteratively selecting the nearest unvisited node, thus minimizing the immediate travel cost at each step. Let the sequence of nodes visited by the greedy solution be denoted as:

$$\sigma = \{0, \sigma_1, \sigma_2, \ldots, \sigma_n, 0\},$$

where 0 is the depot, and $\sigma_1, \sigma_2, \ldots, \sigma_n \in V$ are the nodes visited in the order generated by the Greedy Algorithm.

The total cost of the route produced by the Greedy Algorithm is:

$$C^G = \sum_{i=0}^{n-1} d(\sigma_i, \sigma_{i+1}),$$

where $\sigma_0 = 0$ and $\sigma_{n+1} = 0$ (i.e., returning to the depot).

By definition, C^*, the optimal cost, is the minimum possible cost that visits all nodes in V and returns to the depot:

$$C^* = \min_{\pi \in \Pi} \sum_{i=0}^{n-1} d(\pi_i, \pi_{i+1}),$$

where Π represents the set of all possible permutations of the nodes in V, and π is one such permutation.

Since the Greedy Algorithm does not necessarily minimize the global route cost (it selects locally optimal moves), we know that:

$$C^G \geq C^*.$$

Thus, C^G provides an upper limit on the optimal solution C^*.

B. Minimum Spanning Tree as a Lower Limit

In addition to an upper limit, establishing a lower limit is vital for efficient pruning in QLP. The MST represents the minimal route cost required to connect all nodes, helping QLP identify when extending a route is unproductive and should be pruned.

Let MST denote the weight of the MST of the graph $G = (V, E)$, where V represents the set of delivery nodes and the depot, and E represents the set of edges with distances $d(i, j)$ between nodes i and j. The MST provides a lower limit on the optimal route cost C^* for the TSP:

$$MST \leq C^*.$$

Theorem 2: The weight of the MST provides a lower limit on the optimal route cost C^* for the VRP-UT.

Proof: Let $G = (V, E)$ be a complete undirected graph, and consider the optimal solution to the VRP-UT, denoted by $\mathcal{C}^*$, which is a Hamiltonian cycle that:

- Visits each node in V exactly once.
- Starts and ends at the depot.
- Has a total cost C^*.

The Hamiltonian cycle $\mathcal{C}^*$ is a connected subgraph of G that spans all nodes in V and forms a cycle. By removing any single edge from $\mathcal{C}^*$, we obtain a spanning tree T^* of G, and the weight of this spanning tree satisfies:

$$T^* = C^* - d(i, j),$$

where $d(i, j)$ is the weight of the removed edge from the cycle. Hence, the weight of the spanning tree is always less than or equal to the cost of the Hamiltonian cycle:

$$T^* \leq C^*.$$

Since the MST is the spanning tree with the minimum possible weight, we have:

$$MST \leq T^*.$$

Thus, combining these inequalities, we establish the following relationship:

$$MST \leq C^*.$$

Therefore, the weight of the MST provides a lower limit on the optimal route cost C^*. The MST is computed using Kruskal's Algorithm (see Appendix C), which is known for its efficiency in finding a spanning tree by selecting edges in non-decreasing order of weight.

C. Pruning for Optimal Transfer Efficiency

In VRP-UT with transfer constraints, pruning is essential for reducing computational complexity by eliminating paths that cannot contribute to an optimal solution. The primary objective is to identify transfer scenarios that can be postponed or avoided, enhancing routing efficiency without sacrificing optimality.For clarity, in the following pruning conditions, LLL denotes the minimum required time for completion of a route segment, based on the MST lower bound. Meanwhile, UUU represents an upper bound based on heuristic methods, providing a cap on route costs. These limits guide pruning by helping identify paths that exceed feasible cost thresholds.

- r, r^{rest}: the current incomplete route r and the remaining unvisited nodes for w_1.
- R_{w_i}: the full set of customers associated with warehouse w_i.
- $\mathrm{C}(r)$: the completion time of the current partial route r.
- $\mathrm{L}(r)$ and $\mathrm{U}(r)$: the lower and upper limits on the completion time for route r, derived from MST and Greedy methods, respectively.
- $T^{\text{transfer}}(s_i, s_j)$: the transfer time between stations s_i and s_j, which includes waiting time.

Based on the incomplete route r, we determine whether to prune further exploration or continue with route adjustments. The pruning scenarios below outline how unidirectional and bidirectional transfers impact this decision.

1) Pruning Condition 1: Single-Directional Transfer: This condition applies when transfers are required only from warehouse w_1 to w_2. We prune a partial route r at w_1 if extending it further will not yield an improvement. The pruning condition is met if the following inequalities hold:

$$\mathrm{L}(r^{\text{rest}}) \geq \mathrm{U}(R_{w_2}) + T^{\text{transfer}}(s_1, s_2),$$

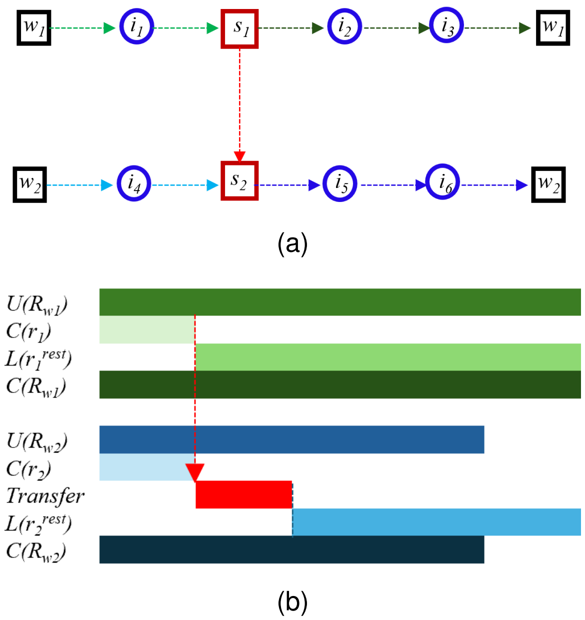

Fig. 3. Example of Pruning Condition 1: Single-Directional Transfer. Subfigure (a) shows the route structure, including partial and rest routes. Subfigure (b) illustrates the time variables in an example case.

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) > \mathrm{U}(R_{w_1}),$$

where r^{rest} represents unvisited customers remaining after visiting station s_1 for transfer. If these conditions hold, the partial route r can be pruned, as extending it would not lead to an optimal solution.

Proof: Consider a scenario where a transfer from w_1 to w_2 is required without a reciprocal transfer. The upper limit of the completion time at w_2 is represented as:

$$C(R_{w_2}) \leq \mathrm{U}(R_{w_2}) + \mathrm{C}(r) + T_{r,s_1} + T^{\text{transfer}}(s_1, s_2).$$

For optimal solution alignment, w_2 should initiate operations in synchronization with the transfer timing. Meanwhile, the lower bound for w_1's completion time is:

$$C(R_{w_1}) \geq \mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}).$$

To prevent suboptimal synchronization, we establish the condition:

$$C(R_{w_1}) \geq C(R_{w_2}),$$

which leads to:

$$\mathrm{L}(r^{\text{rest}}) \geq \mathrm{U}(R_{w_2}) + T^{\text{transfer}}(s_1, s_2).$$

This inequality establishes that w_1's completion time consistently exceeds that of w_2. Consequently, to minimize w_1's completion time, the following must also hold:

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) \leq \mathrm{U}(R_{w_1}).$$

When this condition is not met, the partial route r would exceed $\mathrm{U}(R_{w_1})$, resulting in a suboptimal solution even if it does not affect the total completion time.

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) > \mathrm{U}(R_{w_1}),$$

Thus, this inequality confirms that extending the current partial route r would lead to non-optimal outcomes, justifying pruning. By removing such routes, we focus QLP's exploration on paths that have potential for improved efficiency.

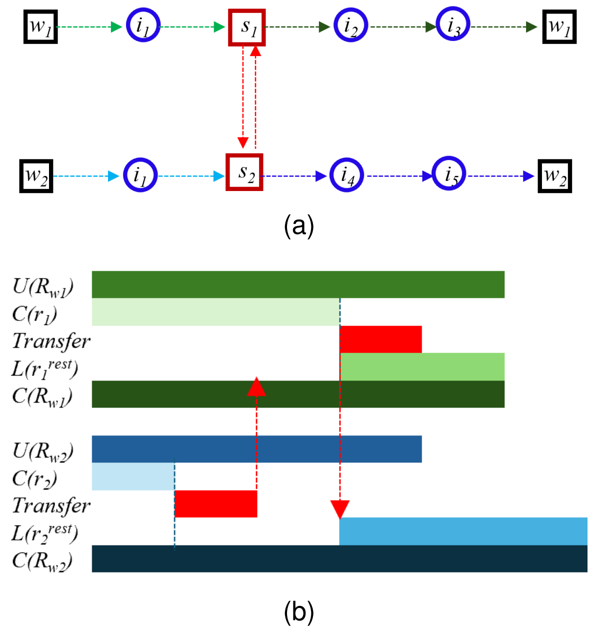

Fig. 4. Example of Pruning Condition 1: Bi-Directional Transfer. Subfigure (a) depicts the route structure having bi-directional movement between stations. Subfigure (b) illustrates the time variables including example case.

2) Pruning Condition 2: Transfer Synchronization Without Unknown Delay: This condition applies when transfers are required in both directions, from s_1 to s_2 and from s_2 to s_1. The aim is to prevent a mismatch in completion times between warehouses that would cause inefficiency due to extended waiting. We prune the current route r if extending it results in delayed transfer synchronization, based on the following inequalities:

$$\mathrm{U}(r^{\text{rest}}) \leq \mathrm{L}(R_{w_2}) + \mathrm{C}(r) + T_{r,s_1},$$

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) > \mathrm{U}(R_{w_1}),$$

Proof: For synchronized completion times, we must ensure that w_1's transfer does not lag w_2's. Thus, we require:

$$C(R_{w_1}) \geq C(R_{w_2}),$$

leading to:

$$\mathrm{L}(r^{\text{rest}}) + T^{\text{transfer}}(s_1, s_2) \geq \mathrm{U}(R_{w_2}) + \mathrm{C}(r) + T_{r,s_1}.$$

By assuming $T^{\text{transfer}}(s_1, s_2) \geq \mathrm{C}(r) + T_{r,s_1}$, we simplify this condition as:

$$\mathrm{L}(r^{\text{rest}}) \geq \mathrm{U}(R_{w_2}) + \mathrm{C}(r) + T_{r,s_1}.$$

Additionally, for the completion time of w_1 to remain below upper limit, the following inequality must hold:

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) \leq \mathrm{U}(R_{w_1}),$$

which ensures that extending route r would exceed w_1's feasible completion time upper bound, $\mathrm{U}(R_{w_1})$, confirming that this path does not contribute to an optimal solution.

$$\mathrm{C}(r) + T_{r,s_1} + \mathrm{L}(r^{\text{rest}}) > \mathrm{U}(R_{w_1}),$$

To maintain optimal timing between w_1 and w_2, this condition ensures that any route that extends beyond this synchronization threshold does not need further exploration, as it cannot contribute to an optimal solution. This pruning criterion prevents inefficiencies caused by mismatched completion times across warehouses.

D. Q-Learning With Pruning

QLP is a reinforcement learning approach tailored to the complexities of the VRP-UT. QLP leverages reinforcement learning to explore feasible routes effectively, incorporating dynamic pruning conditions that reduce the solution space by discarding routes violating feasibility or efficiency constraints.

In each iteration, the agent observes the current state, selects an action, and receives feedback as a reward [65]. This reward updates the Q-table—a matrix storing the utility of state-action pairs—according to the Bellman equation:

$$\begin{aligned} &Q_{t+1}(s,a) \\ &= Q_t(s,a) + \alpha\left[r(s,a) + \gamma \max_{a'} Q(s',a') - Q_t(s,a)\right], \end{aligned}$$

where:

- $Q(s,a)$: the expected utility of taking action a in state s,
- α: the learning rate, controlling the degree to which new information overrides the old,
- $r(s,a)$: the reward received after performing action a in state s,
- s': the state reached after taking action a,
- γ: the discount factor, which determines the weight of future versus immediate rewards.
- $\max_{a'} Q(s',a')$: the maximum Q-value for the next state s', guiding optimal future decisions.

To balance exploration and exploitation, the QLP employs an ϵ-greedy policy in Q-learning. The ϵ-greedy method ensures that the agent has a chance to explore the solution space thoroughly, while gradually focusing on the most promising routes as learning progresses [57]. The value of ϵ decays over time, allowing the agent to prioritize exploitation in later stages.

$$\pi(s) = \begin{cases} a^*, & \text{with probability } 1-\epsilon, \\ a_a, & \text{with probability } \epsilon, \end{cases} \tag{18}$$

where $\pi(s)$ is the decision policy for the current state s, a^* is the best estimated action for the state s at the current time, and a_a is a random action selected with probability ϵ.

In summary, QLP offers an efficient approach to solving the VRP-UT. By integrating two distinct pruning conditions, QLP reduces the solution space dynamically, effectively guiding the Q-learning agent towards optimal routes without extensive exploration. The result is a method that balances efficient decision-making with the flexibility to adapt in uncertain transfer conditions, demonstrating how reinforcement learning can address NP-hard problems with complex routing and timing constraints.

Algorithm 1 Q-Learning With Pruning

```
procedure QLP
    Set the parameters
    Initialize Q-table Q(s, a) = 0 for all state-action pairs
    if transfer required then
        Initialize distance matrices including station
    else
        Build distance matrices excluding station
    end if
    Generate initial solution using a greedy algorithm
    for each warehouse w do
        for each episode t do
            Observe current s
            Select a using ϵ-greedy policy
            Take a and receive immediate r(s, a)
            Observe s′
            if Pruning Condition 1 holds then
                Prune paths
            else if Pruning Condition 2 holds then
                Prune paths
            end if
            Update Q(s, a) using the Bellman equation
            Transition to the new state s′
        end for
    end for
    Calculate travel time and latency for each route
    Update preemption nodes based on pruning decisions
    Output final solution
end procedure
```

V. Numerical Analysis

The computational analysis of the VRP-UT forms a critical component of this study, providing insights into the practical applicability and efficiency of the proposed model. This section details the experimental setup and the resultant analysis obtained from implementing the VRP-UT model and designing heuristics.

A. Experimental Settings

In our research, the experimental setup is meticulously calibrated to capture the intricacies of urban logistics networks. The simulation begins with the strategic placement of warehouses. These are fixed at coordinates (10, 10), (40, 10), and for a scenario involving a third warehouse, at (25, 30), ensuring distinct service areas with a 9km radius to avoid service overlap. The customer delivery points and station coordinates are generated randomly within these predefined radii. 10 replicated instances for each problem size are created to bolster the robustness of our approach against various urban configurations. This randomness in customer locations and their association with specific warehouses is designed to reflect the distribution patterns seen in metropolitan logistics activities.

In each scenario, the VRP-UT model adopts a probabilistic approach to enhance realism in problem instance settings, particularly regarding customer inventory allocation constraints.

TABLE I

DETAIL RESULTS OF THE EXAMPLE OPTIMAL SOLUTION

Vehicle (v)	**Node** (n)	**Designated Warehouse** (W_i)	**Inventory Warehouse** (W_i^{Inv})	**Drop time** ($t^U_{\text{drop},s}$)	**Pickup time** ($t^U_{\text{drop},s}$)	**Arrival time** (t^T_{vi})	**Completion time** (t^T_a)	**Objective value** (C)
0	I4	0	0	-	-	7.43	127.89	127.89
	S7	-	-	80.0	-	80.0		
	I2	0	1	-	-	91.75		
	I6	0	1	-	-	114.95		
	W0	-	-	-	-	127.89		
1	S8	-	-	-	20.0	13.24	69.02	69.02
	I3	1	1	-	-	32.42		
	I5	1	1	-	-	54.20		
	W1	-	-	-	-	69.02		

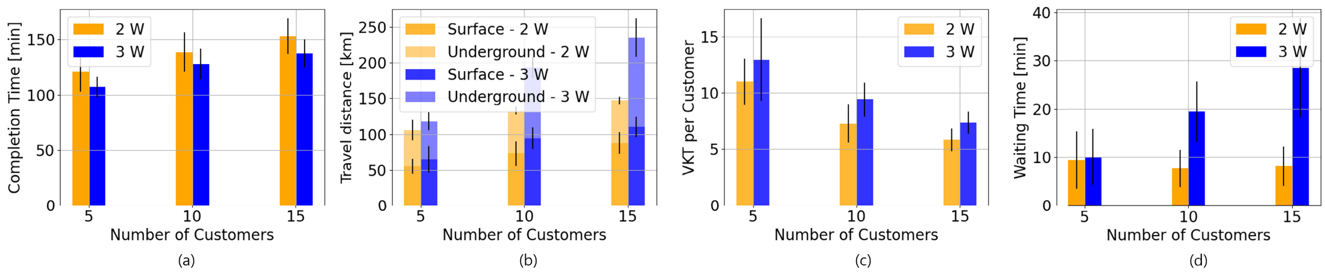

Fig. 5. Computational results of VRP-UT: (a) Completion time, (b) travel distance, (c) VKT per customer, and (d) waiting time.

This method considers a 50% chance that inventory may not be sourced from the customer's nearest warehouse, introducing a variable that accurately mirrors real-world logistics scenarios where stock redistribution is routine. For analytical consistency, travel costs and velocities for vehicles and underground transport are initially set to a unitary rate. This uniformity is purposefully employed to enable a baseline for comparative analyses across different routing strategies. Later, in our sensitivity analysis, these cost and velocity parameters are varied to understand the model's responsiveness to different operational conditions.

The computational experiments are powered by a state-of-the-art processing unit, leveraging the advanced capabilities of the Gurobi Optimizer and Python to ensure that our modeling and algorithmic evaluations are executed within a robust computational framework. Each experiment is constrained by a maximum computation time of 3600 seconds (1 hour) to ensure consistency and practical feasibility across all instances. This setup allows us to accurately assess the efficacy of various heuristic approaches in managing the dynamic and complex environment that is the VRP-UT.

B. Model Verification

The behavior of the VRP-UT model was examined across scenarios involving 5, 10, 15, and 20 randomly distributed customers with either 2 or 3 warehouses. The vehicles and underground transfer system operated at a uniform speed of 30 km/h, and the underground transfers adhered to a fixed schedule, with departures and arrivals at 10-minute intervals.

Figure 6 illustrates the optimal routing solutions for different problem sizes, while Table I presents detailed temporal data for the optimal route from scenario (a) in Figure 6.

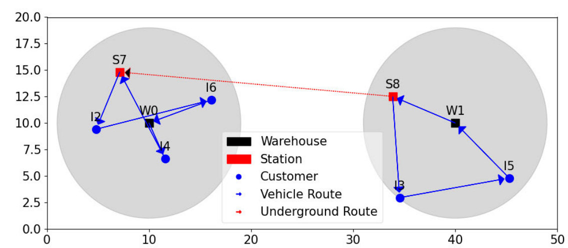

Fig. 6. Optimal solutions from VRP-UT for 2-Warehouse 5-Customers.

In this particular solution, an underground transfer from station 8 to station 7 was necessary, as customers 2 and 6—who were assigned to warehouse 0 ($W_i = 0$)—required products stored in warehouse 1 ($W_i^{\text{Inv}} = 1$). Vehicle 1 transported these products to station 8, where they were dropped off at the underground transfer system at the scheduled drop time ($t^U_{\text{drop},s}$). The goods were then transferred to station 7, where vehicle 0 picked them up at the pickup time ($t^U_{\text{pick},s}$), which coincided with the arrival time of the underground transfer. The vehicle subsequently completed the deliveries to customers 2 and 6.

To rigorously assess the scalable operational impact of ULS on urban traffic congestion, we employed the metric Vehicle Kilometers Traveled (VKT) per customer [12]. This measure directly highlights the traffic contribution of each delivery, offering a precise gauge of congestion potential per service instance. Below is the formal mathematical definition for VKT per customer:

$$\text{VKT per customer} = \frac{\sum_{v \in V} \sum_{(i,j) \in A} D_{ij} \cdot x_{vij}}{|I|} \tag{19}$$

This formula calculates the total vehicle kilometers traveled by all vehicles across all routes, normalized by the number of customers, $|I|$, to reflect the average traffic load each customer contributes to the urban system.

Figure 5 present the computational results of the VRP-UT model across different instance sizes, focusing on completion time, travel distances, VKT per customer, and waiting times.

Across all scenarios, completion times increase as the number of customers grows, with 3-warehouse scenarios generally showing shorter times compared to 2-warehouse cases. This highlights the efficiency of ULS integration, which improves resource allocation and reduces total delivery times. Travel distances remained stable for surface vehicles in 2-warehouse scenarios but increased notably for underground travel in the 3-warehouse cases as customer numbers rose. This suggests that the added complexity of multiple warehouses leads to more underground transfers, increasing operational costs. Meanwhile, VKT per customer decreased by up to 60% as the number of customers increased, indicating the potential for ULS to reduce surface congestion. Waiting time, defined as the time spent waiting for the underground transfer to arrive, was significantly higher in the 3-warehouse scenarios, likely due to the added complexity of coordinating multiple underground transfers.

In addition to the operational metrics, CPU time for the VRP-UT model increased exponentially as the problem size grew, particularly for instances involving 15 customers, where many cases reached the computational limit of 3600 seconds. This highlights the rising computational complexity of the model as more customers and transfers are incorporated. Further details on the computational result are provided in Appendix E.

C. Solution Comparison of VRP-UT With VRP-IC

A key feature of VRP-UT is its underground logistics, which facilitates the exchange and replenishment of inventories, catering to specific customer demands. To evaluate VRP-UT's efficiency, we conducted a comparative analysis with VRP-IC. Unlike VRP-UT, this model excludes underground logistics and focuses on direct service from warehouses while adhering to inventory capacities. VRP-IC operates exclusively on surface networks, not leveraging the operational efficiencies of underground logistics. This model, detailed in Appendix B, requires precise planning and operates exclusively on surface networks, not leveraging the operational efficiencies of underground logistics. Although the specific constraints differ, VRP-IC aligns with variations of Vehicle Routing Problem with Multiple Products and Time Windows (VRP-MPTW), as discussed in research by Coelho and Laporte [31], Kabcome and Mouktonglang [7].

Figure 7 visually compares the optimal routing solutions for VRP-UT and VRP-IC in scenarios with 2 and 3 warehouses serving 10 customers. In the VRP-UT scenarios (Figures 7(a) and 7(c)), vehicles minimize extensive travel outside their designated warehouse areas by utilizing nearby underground stations for transferring goods. This results in shorter vehicle travel paths, as indicated by the blue lines. In contrast, the VRP-IC scenarios (Figures 7(b) and 7(d)) lack the option of underground stations, forcing vehicles to travel longer distances to serve customers outside their warehouse areas. Consequently, the vehicle routes in VRP-IC are significantly longer than those in VRP-UT, underscoring the operational

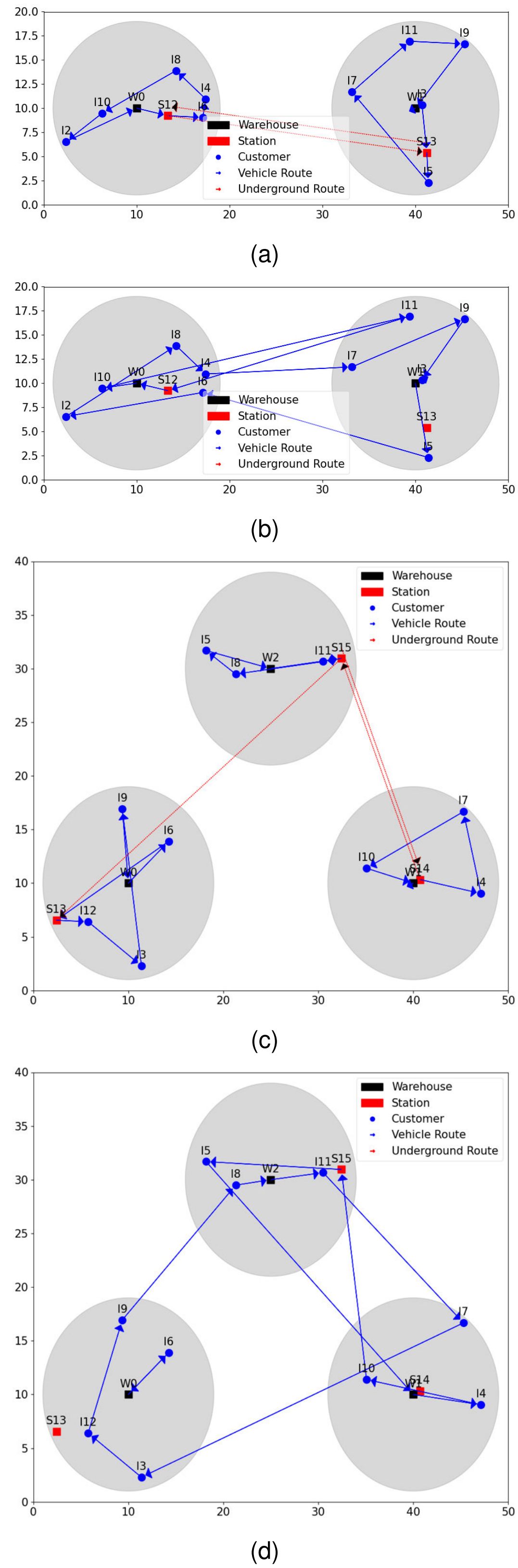

Fig. 7. Optimal solutions 10 customers cases from MILP for 2-Warehouse of (a) VRP-UT, (b) VRP-IC, and 3-Warehouse of (c) VRP-UT, (d) VRP-IC.

TABLE II
PARAMETER SETTINGS FOR HEURISTIC

Algorithm	Parameter	Value
QLP	Learning Rate (α)	0.3
	Discount Factor (γ)	0.9
	Exploration Rate (ϵ)	0.5
	Episodes	500
GA	Population Size	1,000
	Max Generation	1,000
	Tournament Size	5
	Crossover Rate	0.8
	Mutation Rate	0.2
SA	Initial Temperature	100,000
	Cooling Rate	0.995
	Minimum Temperature	1
PSO	Number of Particles	1,000
	Number of Iterations	10,000
	Constant Inertia Weight (ω)	0.5
	Cognitive Constant (c_1)	1.0
	Social Constant (c_2)	2.0

efficiency of integrating underground logistics into urban vehicle routing systems.

The comparative analysis results, illustrated in Figure 8, demonstrate the operational advantages of integrating underground logistics into urban vehicle routing. The VRP-UT model shows superior performance in terms of delivery time, travel distance, and VKT per customer. In terms of time savings, utilizing 2 warehouses results in approximately 20% savings, while 3 warehouses achieve almost 30% savings. In terms of VKT per customer, the VRP-UT model achieves an average reduction of 55%. This significant decrease indicates a substantial reduction in traffic congestion, with surface vehicle travel notably minimized, contributing to less potential traffic buildup. However, as the instance size increases, underground travel distance and waiting time also tend to rise. Therefore, careful assessment of operational conditions is necessary to ensure the economic viability of the system.

Overall, the VRP-UT model not only proves to be operationally efficient but also emerges as a sustainable and strategic solution for urban traffic management. Detailed numerical results from the experiments are provided in Appendix F.

D. Comparison of Heuristic Approaches With Exact Solution

This section provides a comparative evaluation of various heuristic approaches, including the newly proposed QLP, as well as well-established meta-heuristics such as Genetic Algorithm (GA), Simulated Annealing (SA), and Particle Swarm Optimization (PSO). The performance of each heuristic is assessed using its optimality ratio, which measures the deviation from the optimal solution derived through MILP. These heuristics are further compared to the exact solution obtained via the MILP method, solved using the Gurobi commercial solver [17]. The comparison is carried out on instances with up to 15 customers. Table II summarizes the parameter settings that were fine-tuned for each method to ensure optimal performance in solving the VRP-UT. Detailed adaptations of each heuristic for the VRP-UT, along with their corresponding pseudo-codes, can be found in APPENDIX D.

Figure 9 and Table III show the computational results for various problem sizes, focusing on the optimality gap and computational time. Among the heuristic approaches, the QLP approach performs the best overall, with an average optimality gap of 4.47%, closely followed by SA with a gap of 4.51%. GA and PSO show slightly higher optimality gaps compared to QLP and SA. While the differences in optimality are relatively small among these methods, QL-pruning emerges as the superior approach in terms of computational efficiency, achieving substantially shorter computation times. In addition, QL-pruning stands out not only in terms of optimality but also in terms of computational efficiency, significantly reducing the computation time compared to the other methods. Even when compared with the second-fastest heuristic, SA, QL-pruning reduces the computational time by over 99.7.

On the other hand, the exact approach with MILP exhibits an exponential increase in CPU time as the problem size increases, becoming impractical for the 15-customer cases. As a result, in the next section, we compare the performance of heuristics exclusively, as MILP is not feasible for larger instances.

E. Numerical Comparison Between Heuristic Approaches

This section evaluates the performance of various solution approaches when applied to large-scale problem instances, where the number of customers ranges from 5 to 100. The number of warehouses was fixed at either 2 or 3, and all other parameters were consistent with those outlined in earlier sections. Due to the computational limitations of MILP in handling large-scale problems, our analysis focuses on the efficiency of heuristic approaches, specifically QLP, GA, SA, and PSO.

Figure 10 illustrates the results for instances with up to 100 customers, highlighting both the objective values and CPU times for each heuristic. In most cases, the objective value increased with the number of customers, and QLP demonstrated superior performance across most instances. Notably, the performance gap between QLP and other heuristics increased as the problem size grew. On average, QLP outperformed SA by approximately 20%, with the gap widening to 46% in the largest instances. In terms of computational time, QLP showed consistently shorter computation times, except for smaller problem sizes where all heuristics performed similarly. While SA performed marginally faster in some of the larger instances, Q-Learning remained competitive, providing a better balance between solution quality and computational efficiency. As problem sizes increased, other heuristics such as GA and PSO showed a significant rise in both objective value and computational time, especially in instances involving 50 or more customers.

This analysis underscores the scalability and efficiency of Q-Learning with pruning in solving large instances, making it a more practical choice for real-world applications where computational resources are limited and optimal decision-making is required within constrained time frames.

F. Sensitivity Analysis With Case Study

In this sensitivity analysis, we evaluated the impact of varying underground transfer speeds and frequencies on delivery performance. The case study focused on a real-world

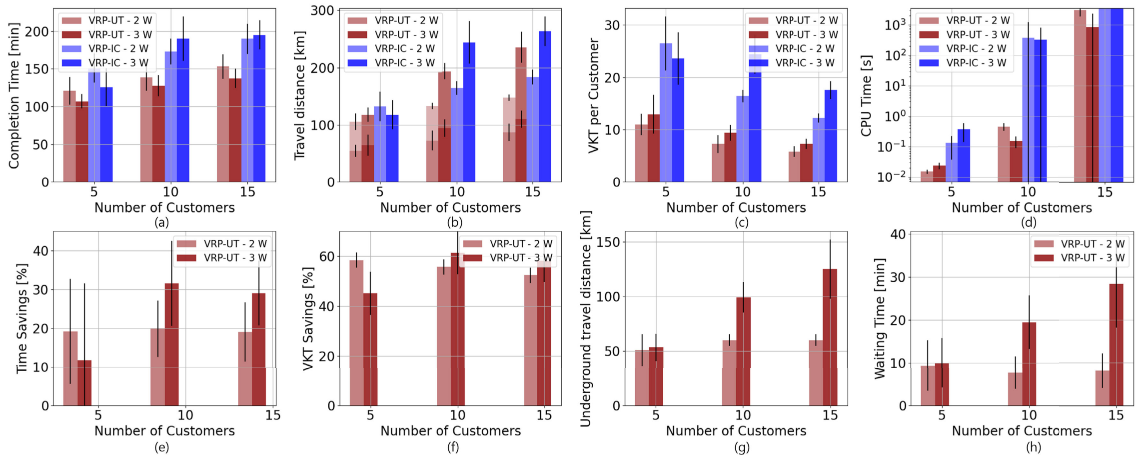

Fig. 8. Computational results of VRP-UT and VRP-IC: (a) Completion time, (b) travel distance, (c) VKT/$|I|$, (d) CPU time, (e) time savings, (f) VKT savings (g) underground travel distance, and (g) waiting time.

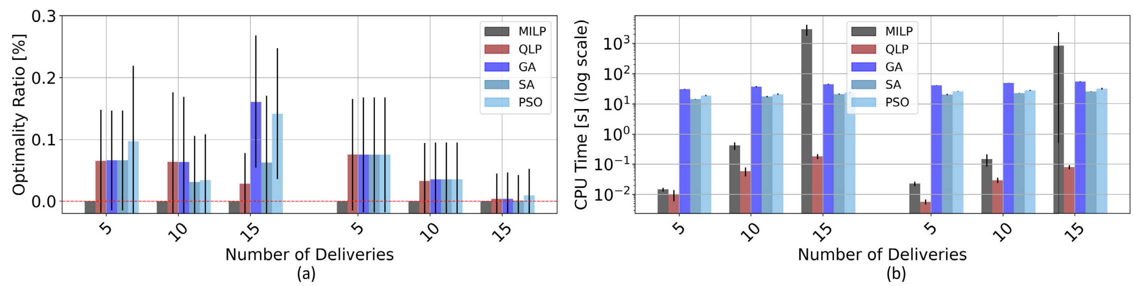

Fig. 9. Comparison of computational results in small instance: (a) optimality, and (b) CPU time.

TABLE III

Computational Results in the VRP-UT Obtained Through MILP, QLP, GA, SA, and PSO

$\|W\|$	$\|I\|$	MILP		QL-pruning			GA			SA			PSO		
		Obj	cpu	Obj	cpu	gap	Obj	cpu	gap	Obj	cpu	gap	Obj	cpu	gap
2	5	120.98	0.01	128.48	0.01	6.49	128.63	30.67	6.60	128.63	14.42	6.60	132.63	18.96	9.72
	10	138.56	0.42	146.71	0.06	6.35	146.70	37.19	6.31	142.50	17.67	3.09	142.93	21.07	3.39
	15	152.67	2949.73	156.60	0.18	2.80	176.85	44.71	16.09	161.83	21.28	6.23	173.67	23.81	14.16
	avg	137.40	983.39	143.93	0.08	5.21	150.73	37.52	9.67	144.32	17.79	5.31	149.74	21.28	9.09
3	5	107.13	0.02	115.14	0.01	7.53	115.30	41.41	7.50	115.30	20.60	7.50	115.30	26.35	7.50
	10	127.88	0.15	132.22	0.03	3.24	132.22	49.78	3.51	132.22	22.99	3.51	132.22	28.29	3.51
	15	137.53	828.68	137.91	0.08	0.38	137.86	53.65	0.37	137.56	25.16	0.12	138.63	31.18	0.91
	avg	124.18	276.28	128.42	0.04	3.72	128.46	48.28	3.80	128.36	22.92	3.71	128.72	28.61	3.97
Overall avg		130.79	629.84	136.18	0.06	4.47	139.60	42.90	6.73	136.34	20.35	4.51	139.23	24.94	6.53

map of Seoul, South Korea, incorporating three underground stations—Ji-Juk, Hakyeoul, and Euljiro 3-ga—aligned with subway line 3. These stations were matched with corresponding warehouse locations: "Seoul Metro Jichuk Train Depot," "Seoul Transportation Corporation Suseo Vehicle Depot," and "Euljiro 3-ga station."

Customers were generated within a 5.31 km radius of each warehouse, ensuring no overlap between regions. For each experiment, a fixed number of 100 customers were simulated, and the underground logistics operations adhered to the subway line 3 weekday schedule, sourced from the Seoul Metropolitan Government's official open data portal [56]. Figure 11 illustrates the geographical locations of the stations and depots used in this case study.

TABLE IV

Case Study Locations of Stations and Depots in Seoul

Type	Location	Coordinates
Stations	Ji-Juk Station	(37.6480, 126.9139)
	Hakyeoul Station	(37.4966, 127.0705)
	Euljiro 3-ga Station	(37.5662, 126.9919)
Depots	Jichuk Train Depot	(37.6523, 126.9059)
	Suseo Train Depot	(37.4812, 127.1102)
	Euljiro 3-ga Station	(37.5662, 126.9919)

For the experiment, we varied the underground transfer speeds between 30 km/h, 60 km/h, 90 km/h, 120 km/h, and 150 km/h. We also varied the frequency of transfers, with

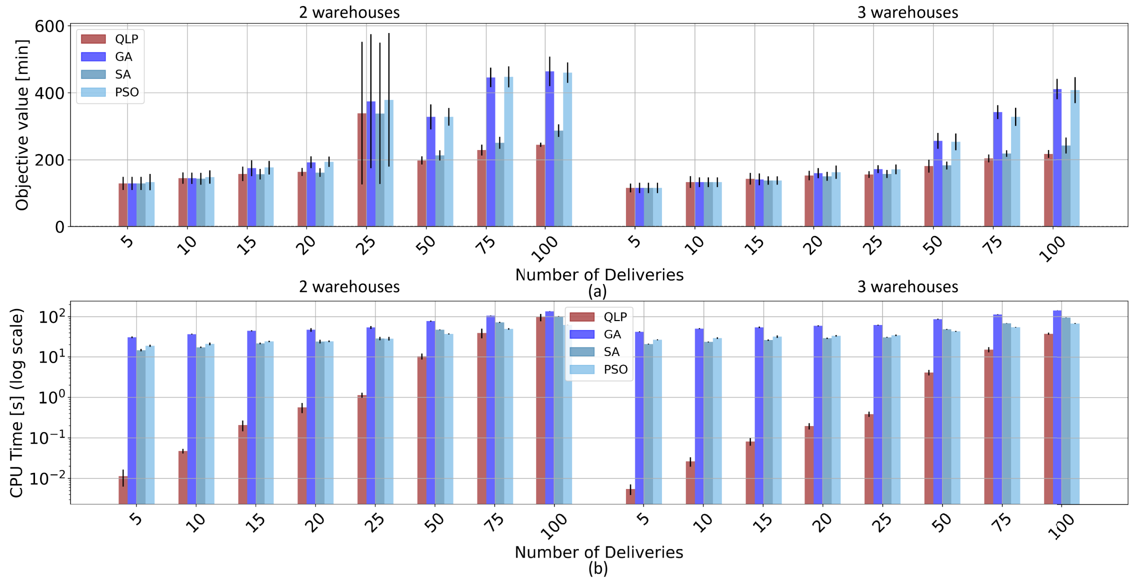

Fig. 10. Comparison of computational results in large size instance: (a) objective value, and (b) CPU time.

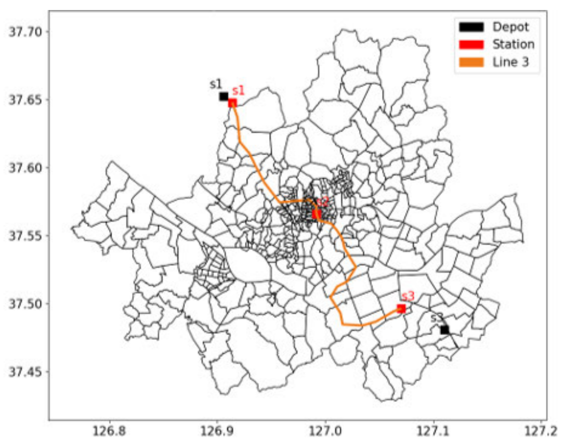

Fig. 11. An illustration of the case study locations of stations and depots within Seoul City, along with Line 3.

intervals of 3, 5, 10, and 15 minutes between schedules. Two scenarios were examined: one involving two warehouses at Ji-Juk and Hakyeoul stations, and the other involving three warehouses utilizing all stations.

As demonstrated in Figure 12 increasing the speed of underground transfers significantly reduces overall completion time, resulting in greater delivery time savings compared to traditional methods. This improvement can be largely attributed to reduced waiting times, as faster transfer speeds decrease overall logistics time. However, while delivery times improved, the variability in VKT remained unpredictable, suggesting that factors other than transfer speed and frequency may influence surface travel distances.

An important aspect of the analysis is the transfer frequency, which is the interval between available underground transfers. The results indicate that as the interval between transfers increases (i.e., lower frequency), delivery times also increase, diminishing the time-saving benefits due to longer waiting times. Interestingly, VKT appears relatively unaffected by changes in transfer frequency, indicating that the distance traveled by surface vehicles is not heavily influenced by underground logistics operations. Detailed sensitivity analysis values are provided in Appendix H.

This analysis is particularly relevant for dense urban areas like Seoul, where underground transportation systems such as the subway are integral to daily commuting and could be leveraged to enhance logistics efficiency. Optimizing subway schedules and speeds can significantly impact last-mile delivery services, reducing both traffic congestion and environmental impact.

G. Management Insights

The results from the sensitivity analysis and case study conducted in Seoul offer key insights for urban logistics managers, transport planners, and policymakers, enabling them to optimize last-mile delivery operations and integrate underground transport systems into urban environments. These findings are particularly valuable for decision-makers looking to improve delivery efficiency while managing the limitations of urban infrastructure.

Based on the sensitivity analysis and case study results, we can derive several actionable insights for urban logistics practitioners and policymakers:

1) **Transfer Speed and Delivery Efficiency**: Increasing the speed of underground transfers significantly reduces delivery times. For example, when the transfer speed increases from 30 km/h to 150 km/h, delivery time is reduced by approximately 29%. In addition, time savings compared to traditional truck delivery increase from 27% to 49%, demonstrating substantial efficiency improvements. For logistics companies, this highlights the importance of collaborating with public transport authorities or investing in infrastructure to support faster underground transfer speeds. Faster transfers not only

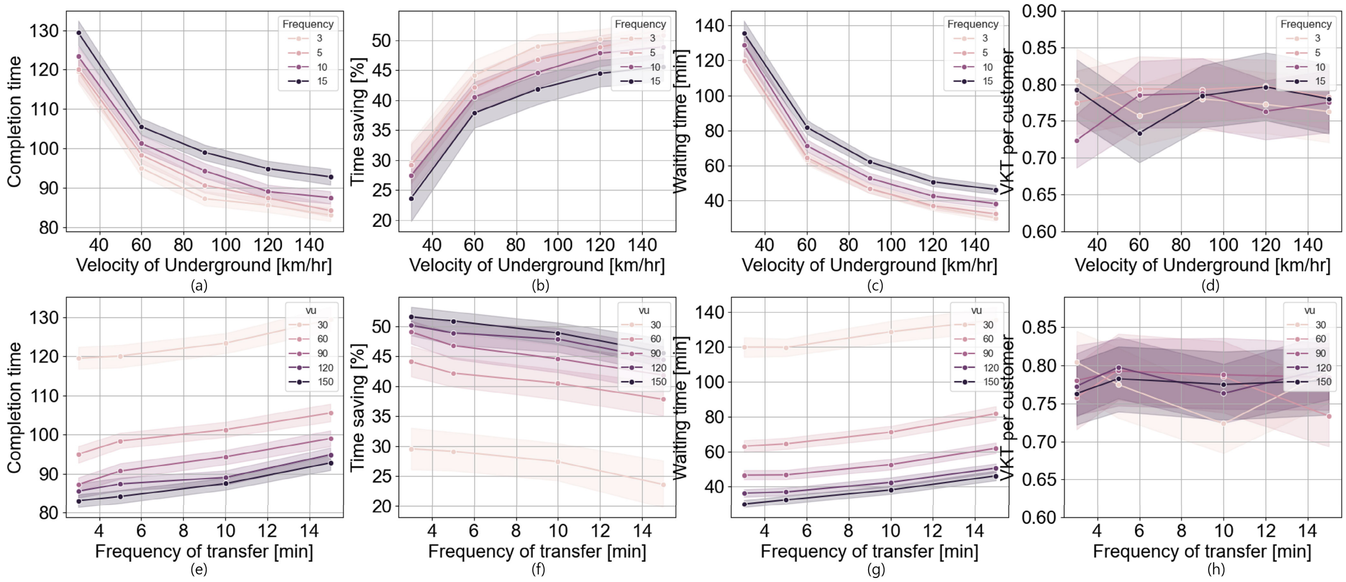

Fig. 12. Results of the sensitivity analysis: (a) Completion time, (b) Time savings, (c) Waiting time, (d) VKT per customer with varying velocities of underground transfer, and (e) Completion time, (f) Time savings, (g) Waiting time, (h) VKT per customer with varying frequencies of underground transfer.

help alleviate congestion but also improve delivery times and lower operational costs, which can give companies a competitive advantage in the market.

2) **Impact of Transfer Frequency**:
The frequency of underground transfers also plays a crucial role in improving delivery efficiency. When the interval between transfers is reduced from 15 minutes to 3 minutes, delivery times decrease by 9.8%, and time savings increase by 16.1%. This shows that more frequent transfers can significantly speed up deliveries and reduce customer waiting times. Urban logistics managers should focus on ensuring high-frequency underground transfers, particularly during peak hours. Collaboration with public transport authorities to align delivery schedules with available underground transfers will be key to maximizing delivery performance, especially in crowded urban areas.

3) **Scalability and Expansion Potential**:
Expanding the number of integrated underground stations from two to three results in a 7.5% reduction in delivery time and a 70% increase in time savings. This demonstrates the scalability potential of underground logistics systems. For both logistics companies and city planners, this means that investing in expanding underground logistics infrastructure can significantly enhance delivery efficiency. While adding more stations may slightly increase waiting times, the overall impact on delivery times remains highly beneficial. Urban planners and logistics companies should advocate for long-term collaboration between public and private sectors to integrate logistics operations into the growing underground transport networks, ensuring that infrastructure expansion keeps pace with rising demand for faster deliveries.

4) **Limited Impact on Surface Travel Distance**:
Despite improvements in underground transfer speed and frequency, surface travel distances (measured as Vehicle-Kilometers Traveled, or VKT) remain largely unaffected. This suggests that while underground transfer efficiency improves, other factors, such as traffic conditions and surface road infrastructure, still play a significant role in determining overall delivery performance. Urban logistics managers need to account for these factors when planning delivery routes and integrating underground logistics with existing surface transportation systems. This insight emphasizes the need for a holistic approach to logistics that considers both underground and surface-level transportation challenges.

These insights offer practical guidance for urban logistics managers and city planners. By leveraging underground transport systems, logistics companies can reduce operational costs, improve delivery speeds, and enhance the efficiency of last-mile delivery. Additionally, policymakers can use these findings to plan for scalable infrastructure that meets the growing demand for urban deliveries while reducing the negative impacts of congestion and improving sustainability. By focusing on the optimization of transfer speeds, frequencies, and system scalability, urban areas can create more efficient, sustainable logistics systems that are better suited to the demands of modern urban life.

VI. Conclusion and Discussion

This study introduced the VRP-UT, a novel model that integrates traditional vehicle routing with ULS to address the challenges of urban congestion, operational efficiency, and sustainability. We explored the impact of incorporating ULS, particularly the use of real-world subway systems and schedules, into delivery logistics. Our model was further enhanced by the introduction of a novel QLP approach, designed to optimize route decisions and streamline operations by reducing unnecessary transfers.

Our findings demonstrate the significant potential of the VRP-UT model to reduce surface-level congestion by diverting a considerable portion of delivery traffic underground. Specifically, the model achieved a reduction in VKT by approximately 50% compared to traditional vehicle routing models such as VRP-IC. The use of underground links for inventory transfers alleviated surface-level congestion, particularly in densely populated areas. Moreover, the model showed notable operational cost savings, especially as the number of warehouses increased, highlighting its economic viability for urban logistics. The inclusion of real subway schedules into the model ensured that underground transfers were synchronized with surface deliveries, improving overall logistics coordination. Sensitivity analyses revealed that faster underground transfer velocities could reduce completion times by up to 50%, further improving delivery speed and reducing congestion.

We also proposed a novel QLP approach that dynamically reduces the search space by eliminating unnecessary operations. This approach was validated through comparisons with traditional heuristics like GA, SA, and PSO, consistently achieving superior optimality while maintaining lower computational time. The Q-learning-based method proved effective in managing the complexity introduced by underground transfer schedules and multiple delivery nodes. A comprehensive case study based on Seoul's subway Line 3 was conducted to validate the model's practicality. By incorporating actual subway maps, depot locations, and scheduling data, the case study provided insights into the real-world applicability of VRP-UT, offering practical results on how underground logistics can integrate with traditional vehicle routes.

While this research makes important contributions, certain limitations must be acknowledged. The model assumes fixed underground schedules and does not account for real-time traffic variability or disruptions in subway services. Future research could explore the dynamic adaptation of schedules using real-time data, as well as the integration of other emerging technologies like aerial drones and autonomous vehicles to complement underground logistics. Further research could also investigate the environmental impacts of underground logistics, focusing on emission reductions and energy savings, along with the policy frameworks needed for ULS implementation. Additionally, exploring the role of advanced AI and IoT technologies in enhancing the intelligence and efficiency of underground logistics systems would be a valuable avenue for future investigation.

In conclusion, this study represents a transformative approach to urban logistics, offering both theoretical advancements through the VRP-UT model and practical insights through real-world validation. The model aligns with sustainable urban development goals and provides a robust framework for addressing the growing challenges of last-mile delivery in densely populated urban areas.

Appendix

A. Greedy Algorithm

The Greedy Algorithm for VRP-UT lays the foundation for initial solution generation. For classical VRPs with time windows, the greedy algorithm has been shown to provide improved initial feasible solutions [5], [18], [27]. Given VRP-UT's specific feasibility constraints on pickups and deliveries, the algorithm is designed to visit the station whenever drop-off or pickup is necessitated, followed by the construction of routes by choosing the nearest unvisited delivery point from each warehouse, focusing on minimizing immediate travel costs and times. This simple yet effective algorithm provides a solid base for more complex heuristic algorithms by offering a practical starting solution that aligns with the unique constraints of VRP-UT.

Algorithm 2 Greedy Algorithm

```
procedure GREEDY
    Initialize by loading input data
    for each warehouse w ∈ W do
        Define customers/pickup/drop for warehouse w
        Initialize route for warehouse w
        while there exist unvisited customers do
            Update route with nearest remaining customer
        end while
        Deconstruct the first visit
        Reconstruct the route with a station visit
        Update total route to return to warehouse
    end for
    Combine routes from all warehouses
    Calculate total operational cost and completion time
    Save and return results
end procedure
```

B. Vehicle Routing Problem With Inventory Constraints

$$\text{Minimize}: \sum_{v\in V}\sum_{(i,j)\in A} c_T \cdot D_{ij} \cdot x_{vij} \tag{20}$$

$$\text{Subjected to } \sum_{i\in N}\sum_{v\in V} x_{vij} = 1 \quad \forall j \in I \tag{21}$$

$$\sum_{w\in W}\sum_{j\in N} x_{vwj} \leq 1 \quad \forall v \in V \tag{22}$$

$$\sum_{j\in N} x_{vW_vj} = 1 \quad \forall v \in V \tag{23}$$

$$\sum_{j\in N} x_{vjW_v} = 1 \quad \forall v \in V \tag{24}$$

$$\sum_{j\in N} x_{vji} = \sum_{j\in N} x_{vij} \quad \forall v \in V, \forall i \in N \tag{25}$$

$$t^T_{vj} \geq t^T_{vi} + D_{ij} \text{ if } x_{vij} = 1, \forall v \in V, \forall (i,j) \in A \tag{26}$$

$$t^T a_v \geq t^T_{vi} + D_{iw} \quad \text{if } x_{viw} = 1, \forall v \in V, \forall i \in N \tag{27}$$

$$\sum_{j\in N} x_{vij} = 1 \ \forall v \in V, \forall i \in I, \forall w = W_i^{Inventory} \tag{28}$$

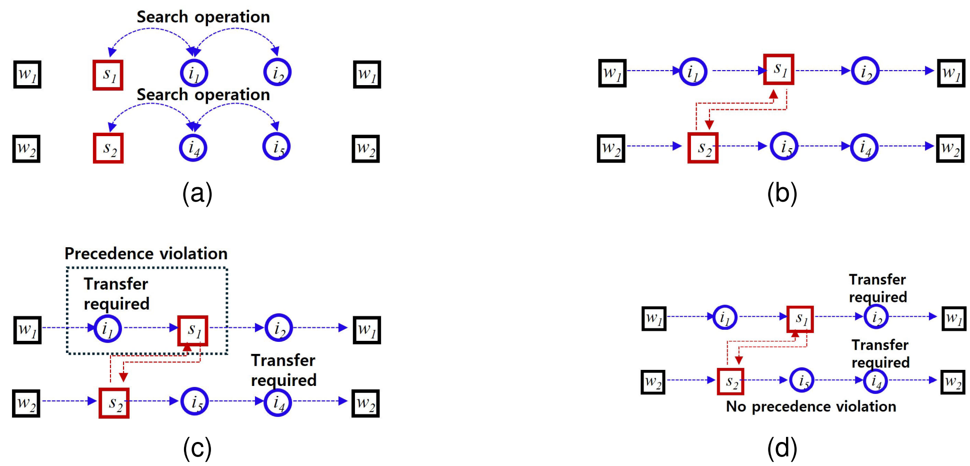

Fig. 13. Illustration of the solution configuration for VRP-UT using heuristic algorithms with a penalty approach: (a) searching for solutions with each search operator, (b) constructing underground travel, (c) a case where the precedence of transfer-required customer and station is violated, (d) a case without precedence violation.

Algorithm 3 Kruskal's Algorithm

```
procedure KruskalMST(graph)
    Initialize an empty graph T for the MST
    Create a list of edges and sort in non-decreasing order
    Initialize a union-find data structure to track components
    for each edge (u, v) in sorted edges do
        if u ≠ v then
            Add edge (u, v) to T
            Union(u, v)
        end if
    end for
    return the graph T as the MST
end procedure
```

Algorithm 4 Genetic Algorithm for VRP-UT

```
procedure GA
    Initialize by loading input data and GA parameters
    for each warehouse w ∈ W do
        Define customers/pickup/drop for this warehouse
        Define genetic representation and evaluation
        Generate initial population including greedy
        for a set number of generations do
            Evaluate fitness with penalty
            Perform Tournament Selection
            Apply crossover with crossover rate
            Apply mutation with mutation rate
            Update population with new generation
        end for
        Store the best solution found
    end for
    Combine optimized routes from all warehouses
    Calculate total operational cost and completion time
    Save and return results
end procedure
```

C. Kruskal's Algorithm for MST

Kruskal's Algorithm is a greedy method for finding the Minimum Spanning Tree (MST) of a connected, undirected graph. It begins by initializing a forest where each vertex is a separate tree and creates a sorted list of all edges based on their weights. The algorithm repeatedly selects the smallest edge that connects two different trees, adding it to the MST and merging the trees, until $V-1$ edges are included, where V is the number of vertices.

The time complexity of Kruskal's Algorithm is dominated by the edge sorting step, resulting in $O(E \log E)$ or $O(E \log V)$, where E is the number of edges. Union-find operations can be performed in nearly constant time, making the overall complexity efficient. Compared to NP-hard problems like the Traveling Salesman Problem (TSP), Kruskal's Algorithm offers a simpler and more efficient approach, making it suitable for various applications in network design and clustering.

D. Heuristic Approaches

The VRP-UT extends the classical multi-depot VRP by incorporating inventory constraints and time windows, introducing additional complexity by managing pickups, drops, and transfers across underground stations. Considering the NP-hard nature of the standard VRP, the VRP-UT significantly escalates in complexity. This section introduces algorithms tailored to mitigate the complexity of VRP-UT, including the Greedy Algorithm for initial solution construction and well-known search-based metaheuristic algorithms—GA, SA, and PSO [2], [33], [35].

GA, SA, and PSO are well-known to be efficient solution approaches that successfully address the high complexity of different optimization problems. These algorithms are often compared due to their well-known efficiency and distinctive characteristics [13], [14], [67]. They can be applied to optimization problems by tailoring the solution representations, search operators, and evaluation functions.

To tailor these heuristics for VRP-UT, we strategically partition the problem based on individual warehouses, accommodating inventory discrepancies that require specific pickups or drop-offs. By addressing these

TABLE V
COMPUTATIONAL RESULTS IN THE VRP-UT OBTAINED THROUGH MILP

$\|W\|$	$\|I\|$	Completion Time		Vehicle Travel Distance		Underground Travel Distance		VKT/$\|I\|$		Waiting Time		CPU Time	
		Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
2	5	120.98	18.14	55.00	10.23	50.85	14.51	11.00	2.05	9.31	5.92	0.01	0.00
	10	138.56	17.80	72.94	17.17	59.95	5.36	7.29	1.72	7.66	3.85	0.45	0.13
	15	153.19	16.03	87.49	15.25	59.95	5.36	5.83	1.02	8.11	4.07	3086.29	1180.49
3	5	107.13	9.30	64.71	18.38	53.31	12.25	12.94	3.68	9.98	5.80	0.02	0.01
	10	127.88	13.93	94.26	15.07	99.12	14.07	9.43	1.51	19.41	6.25	0.15	0.06
	15	137.53	12.76	110.20	14.67	124.97	26.97	7.35	0.98	28.43	10.29	836.91	1460.40

TABLE VI
COMPARATIVE RESULTS OF VRP-IC AND VRP-UT

Parameters		**VRP-IC**				**VRP-UT**							
$\|W\|$	$\|I\|$	Obj	V travel	VKT/$\|I\|$	CPU	Obj	V travel	U travel	VKT/$\|I\|$	Waiting Time	Time Savings	VKT/$\|I\|$ Savings	CPU
2	5	151.01	132.25	26.45	0.13	120.98	55.00	50.85	11.00	9.31	19.17	57.73	0.01
	10	172.92	164.47	16.45	378.56	138.56	72.94	59.95	7.29	7.66	19.83	55.96	0.45
	15	189.85	183.54	12.24	3600.03	153.19	87.49	59.95	5.83	8.11	19.00	52.53	3086.29
3	5	125.73	117.87	23.57	0.37	107.13	64.71	53.31	12.94	9.98	11.76	42.88	0.02
	10	190.16	243.62	24.36	333.28	127.88	94.26	99.12	9.43	19.41	31.59	60.78	0.15
	15	195.05	263.33	17.56	3600.02	137.53	110.20	124.97	7.35	28.43	29.05	58.14	836.91

Algorithm 5 Simulated Annealing for VRP-UT

procedure SA
  Initialize by loading input data and SA parameters
  for each warehouse $w \in W$ **do**
    Define customers/pickup/drop for this warehouse
    Generate initial greedy solution
    Set current solution to greedy solution
    Set initial temperature
    while temperature > minimum temperature **do**
      Generate neighboring solution
      Evaluate neighbor solution with penalty
      Apply acceptance criteria
      if neighbor solution is improved **then**
        Update current solution and cost
      end if
      Reduce temperature with cooling rate
    end while
  end for
  Combine optimized routes
  Calculate total operational cost and completion time
  Save and return results
end procedure

Algorithm 6 Particle Swarm Optimization for VRP-UT

procedure PSO
  Initialize by loading input data and PSO parameters
  for each warehouse $w \in W$ **do**
    Define customers/pickup/drop for this warehouse
    Initialize particles with random routes
    Include a greedy solution in initial particle set
    for a set number of iterations **do**
      for each particle **do**
        Update velocity based on ω, $c1$, and $c2$
        Update position
        Evaluate new position with penalty
        if new position is improved **then**
          Update particle's best position
          Update the global best position
        end if
      end for
    end for
  end for
  Combine optimized routes
  Calculate total operational cost and completion time
  Save and return results
end procedure

segments, the heuristics coordinate the pickup and delivery operations within the constraints of inventory and timing.

Figure 13 illustrates the heuristic algorithms' solution construction sequence with the penalty approach. Initially, each heuristic's search operator adapts to find solutions. Then, underground travel is constructed to accommodate inventory discrepancies. During the evaluation, solutions that violate the precedence of the visiting sequence for customers requiring transfer from other warehouses incur penalties in the objective value, steering the algorithms towards feasible solutions [1], [4], [15].

1) Genetic Algorithm (GA): The GA is tailored explicitly to VRP-UT's unique constraints, including mandatory station visits, pickups, drop-offs, deliveries, and precedence. The algorithm initiates by partitioning the problem based on different warehouses. Each segment generates an initial population comprising both randomly generated and greedy algorithm-derived solutions. Incorporating a penalty function, the GA evaluates the fitness of each individual. Tournament selection strategy [54] is employed to identify superior solutions, while two-point crossover [59] and swap mutation operators [59] are applied to introduce variations. These processes balance exploring new solutions and exploiting efficient existing routes. The population is continuously updated with new generations, cycling through evaluation until reaching the final generation. After determining optimized routes for each

TABLE VII
COMPUTATIONAL RESULTS FOR DIFFERENT ALGORITHMS

$\|W\|$	$\|I\|$	QL-pruning		GA		SA		PSO	
		obj	cpu	obj	cpu	obj	cpu	obj	cpu
2	5	128.48	0.01	128.63	30.41	128.63	14.64	132.63	18.72
	10	144.31	0.05	144.03	36.99	142.50	17.19	147.53	20.92
	15	156.78	0.21	174.46	45.14	156.19	21.38	176.27	23.99
	20	162.94	0.57	192.46	47.69	161.01	23.83	194.11	24.14
	25	339.05	1.14	374.45	54.49	338.40	28.32	378.73	28.15
	50	197.99	10.34	328.32	78.12	213.54	47.78	328.46	37.77
	75	229.35	39.70	446.00	106.44	251.09	72.83	448.01	50.23
	100	245.17	97.44	464.01	136.70	287.37	103.01	460.41	62.70
3	5	115.14	0.01	115.30	42.54	115.30	20.67	115.30	26.36
	10	132.53	0.03	132.22	51.02	132.22	23.34	132.22	29.03
	15	142.41	0.08	140.54	55.04	137.36	25.91	137.31	31.46
	20	152.23	0.20	159.25	59.83	149.62	28.81	162.03	33.17
	25	155.02	0.39	171.04	62.42	156.85	30.25	170.51	33.89
	50	180.52	4.11	256.99	87.66	182.42	49.19	253.97	43.68
	75	204.51	15.11	342.63	113.86	219.19	68.91	328.61	55.20
	100	217.79	37.98	411.23	142.90	242.34	94.83	407.87	67.77

TABLE VIII
SENSITIVITY ANALYSIS RESULTS FOR VARYING UNDERGROUND TRANSFER SPEEDS AND FREQUENCIES

$\|W\|$	$\|v_U\|$	frequency	Obj		Time Saving		Waiting Time		VKT/$\|I\|$	
			avg	std	avg	std	avg	std	avg	std
2	30	3	127.80	11.58	14.11	8.00	111.34	19.02	0.68	0.16
		5	129.11	11.73	13.21	8.43	114.86	19.33	0.66	0.16
		10	130.54	9.83	12.30	6.56	117.68	20.23	0.66	0.16
		15	139.54	12.43	6.19	8.92	122.51	21.27	0.68	0.16
	60	3	100.28	10.26	32.60	7.06	55.95	9.81	0.66	0.16
		5	103.16	8.84	30.65	6.41	57.48	9.85	0.67	0.16
		10	106.19	9.30	28.65	6.22	66.93	10.70	0.68	0.16
		15	112.61	9.75	24.36	6.33	73.65	12.26	0.63	0.16
	90	3	87.76	7.71	41.04	5.06	39.33	6.95	0.66	0.16
		5	94.00	9.72	36.83	6.65	41.05	6.97	0.65	0.16
		10	98.72	9.63	33.65	6.74	45.02	7.74	0.67	0.16
		15	103.67	8.83	30.34	6.09	54.75	9.55	0.68	0.16
	120	3	84.40	7.51	43.28	5.16	30.63	5.38	0.67	0.16
		5	88.62	7.97	40.44	5.55	31.62	5.52	0.70	0.16
		10	91.36	8.79	38.62	5.91	35.56	6.25	0.67	0.16
		15	96.90	9.15	34.88	6.34	44.33	8.17	0.66	0.16
	150	3	82.19	6.89	44.78	4.58	24.56	4.30	0.65	0.16
		5	83.98	8.05	43.56	5.55	26.24	4.24	0.66	0.16
		10	88.04	8.15	40.82	5.81	31.52	5.41	0.65	0.16
		15	95.69	8.99	35.70	6.19	40.48	6.85	0.63	0.16
3	30	3	111.19	13.08	45.03	6.53	128.72	34.45	0.93	0.21
		5	111.01	10.93	45.10	5.65	124.93	29.65	0.89	0.21
		10	116.15	11.63	42.56	5.90	140.06	35.98	0.79	0.19
		15	119.21	11.30	41.07	5.67	148.85	40.79	0.91	0.21
	60	3	89.75	8.62	55.61	4.51	70.41	18.36	0.86	0.21
		5	93.58	9.54	53.70	5.08	71.66	18.90	0.92	0.21
		10	96.39	8.21	52.35	4.05	75.62	20.30	0.89	0.22
		15	98.41	7.75	51.33	4.05	90.02	23.29	0.84	0.21
	90	3	86.80	10.68	57.04	5.76	53.83	15.15	0.90	0.22
		5	87.47	9.68	56.72	5.21	52.38	16.49	0.93	0.21
		10	89.98	7.24	55.53	3.36	60.33	16.87	0.91	0.22
		15	94.35	9.31	53.33	4.85	69.26	17.74	0.89	0.23
	120	3	86.77	9.22	57.05	5.14	41.84	12.26	0.87	0.22
		5	86.25	9.61	57.31	5.24	42.25	12.78	0.89	0.22
		10	86.83	8.31	57.07	4.21	49.31	15.67	0.86	0.21
		15	92.87	9.42	54.07	4.84	56.88	15.78	0.93	0.22
	150	3	84.06	9.56	58.40	5.20	35.37	12.09	0.87	0.21
		5	84.48	8.74	58.20	4.68	38.43	12.90	0.90	0.22
		10	87.00	8.41	56.96	4.53	44.71	13.92	0.90	0.21
		15	89.92	9.43	55.54	4.79	51.86	16.05	0.93	0.22

warehouse, the algorithm constructs the underground travel segments by combining these routes. Finally, it calculates the total operational cost and completion time, concluding by returning the results.

2) Simulated Annealing (SA): Similarly, SA starts with initialization and problem partitioning. Each warehouse segment begins with a greedy solution and then methodically explores neighboring solutions to enhance routing. Governed by a carefully designed cooling schedule, SA effectively avoids local optima, as detailed in the study by Vincent et al. [70]. This approach allows for a comprehensive solution space search, with minor variations in route plans being systematically evaluated. New solutions are accepted based on specific criteria, enabling the algorithm to uncover potentially global optimal solutions. SA concludes its process by combining optimized routes from each warehouse and calculating the total operational cost and completion time.

3) Particle Swarm Optimization (PSO): PSO utilizes swarm intelligence principles, where each particle in the swarm represents a potential solution initially derived from a greedy algorithm. As highlighted in the work of Marinakis et al. [44], the particles adjust their paths based on individual and collective experiences. This balance between exploration and exploitation is crucial for PSO's effectiveness. The algorithm evaluates multiple routes simultaneously, considering the complex interplay between surface and underground station logistics. Through iterative updates of particle positions, PSO dynamically searches for optimal solutions within the VRP-UT framework, concluding with aggregating these routes and calculating total operational costs and completion times.

E. Computational Results in the VRP-UT Obtained Through MILP

See Table V.

F. Comparative Results of VRP-IC and VRP-UT

See Table VI.

G. Computational Results From Heuristic for Large Size Instances

See Table VII.

H. Sensitivity Analysis Results

See Table VIII.

References

[1] M. Agarwal and R. Gupta, "Penalty function approach in heuristic algorithms for constrained redundancy reliability optimization," *IEEE Trans. Rel.*, vol. 54, no. 3, pp. 549–558, Sep. 2005.

[2] A. Askarzadeh, L. dos Santos Coelho, C. E. Klein, and V. C. Mariani, "A population-based simulated annealing algorithm for global optimization," in *Proc. IEEE Int. Conf. Syst., Man, Cybern. (SMC)*, Oct. 2016, pp. 004626–004633.

[3] A. Van Binsbergen and P. Bovy, "Underground urban goods distribution networks," *Innovation: Eur. J. Social Sci. Res.*, vol. 13, no. 1, pp. 111–128, Mar. 2000.

[4] R. H. Byrd, J. Nocedal, and R. A. Waltz, "Steering exact penalty methods for nonlinear programming," *Optim. Methods Softw.*, vol. 23, no. 2, pp. 197–213, Apr. 2008.

[5] L. B. Cavalcanti and A. B. Mendes, "A greedy heuristic for the vehicle routing problem with time windows, synchronization constraints and heterogeneous fleet," in *Proc. 10th Int. Conf. Comput. Logistics*, Barranquilla, Colombia. Cham, Switzerland: Springer, Sep. 2019.

[6] Z. Chen, J. Dong, and R. Ren, "Urban underground logistics system in China: Opportunities or challenges?" *J. Underground Space*, vol. 2, no. 3, pp. 195–208, Sep. 2017.

[7] L. C. Coelho and G. Laporte, "A branch-and-cut algorithm for the multi-product multi-vehicle inventory-routing problem," *Int. J. Prod. Res.*, vol. 51, nos. 23–24, pp. 7156–7169, Nov. 2013.

[8] J. Cui and J. D. Nelson, "Underground transport: An overview," *Tunnelling Underground Space Technol.*, vol. 87, pp. 122–126, May 2019.

[9] L. Dablanc, "Goods transport in large European cities: Difficult to organize, difficult to modernize," *Transp. Res. A, Policy Pract.*, vol. 41, no. 3, pp. 280–285, Mar. 2007.

[10] J. Dong, W. Hu, S. Yan, R. Ren, and X. Zhao, "Network planning method for capacitated metro-based underground logistics system," *Adv. Civil Eng.*, vol. 2018, no. 1, Jan. 2018, Art. no. 6958086.

[11] *Can Drones Be Tracked?* Droneblog, Houghton, NY, USA, 2022. [Online]. Available: https://www.droneblog.com/can-drones-be-tracked

[12] G. Duranton and M. A. Turner, "The fundamental law of road congestion: Evidence from U.S. cities," *Amer. Econ. Rev.*, vol. 101, no. 6, pp. 2616–2652, Oct. 2011.

[13] R. C. Eberhart and Y. Shi, "Comparison between genetic algorithms and particle swarm optimization," in *Evolutionary Programming VII*. Cham, Switzerland: Springer, 1998, pp. 611–616.

[14] Y. Gao and S. Xie, "Particle swarm optimization algorithm based on simulated annealing (sa) approach," *Comput. Eng. Appl.*, vol. 40, pp. 47–49, Jan. 2004.

[15] J. D. Griffin and T. G. Kolda, "Nonlinearly constrained optimization using heuristic penalty methods and asynchronous parallel generating set search," in *Applied Mathematics Research eXpress*. Oxford, U.K.: Oxford Univ. Press, 2010, pp. 36–62.

[16] D. Guo, Y. Chen, J. Yang, Y. H. Tan, C. Zhang, and Z. Chen, "Planning and application of underground logistics systems in new cities and districts in China," *Tunnelling Underground Space Technol.*, vol. 113, Jul. 2021, Art. no. 103947.

[17] Gurobi Optim., LLC. (2023). *Gurobi Interactive Shell (win64), Version 11.0.0*. [Online]. Available: https://www.gurobi.com/

[18] Y. He, M. Qi, F. Zhou, and H. Li, "Performance analysis of greedy-based construction heuristics on classical vehicle routing problem," in *Proc. IEEE Int. Conf. Ind. Eng. Eng. Manage. (IEEM)*, Dec. 2020, pp. 591–594.

[19] L. Henry, J. Q. Cui, D. Lin, and D. J. Guo, "Feasibility of underground freight transport in New York city and lessons learned and implications to other major cities in the world," *Mod. Transp. Technol.*, vol. 5, no. 4, pp. 86–92, 2008.

[20] W. Hu, J. Dong, and N. Xu, "Multi-period planning of integrated underground logistics system network for automated construction-demolition-municipal waste collection and parcel delivery: A case study," *J. Cleaner Prod.*, vol. 330, Jan. 2022, Art. no. 129760.

[21] W. Hu, J. Dong, K. Yang, B.-G. Hwang, R. Ren, and Z. Chen, "Modeling real-time operations of metro-based urban underground logistics system network: A discrete event simulation approach," *Tunnelling Underground Space Technol.*, vol. 132, Feb. 2023, Art. no. 104896.

[22] S. Hulagu and H. B. Celikoglu, "Electrified location routing problem with energy consumption for resources restricted archipelagos: Case of Buyukada," in *Proc. Forum Integr. Sustain. Transp. Syst. (FISTS)*, Nov. 2020, pp. 323–327, doi: 10.1109/FISTS46898.2020.9264865.

[23] S. Hulagu and H. B. Celikoglu, "Electric vehicle location routing problem with vehicle motion dynamics-based energy consumption and recovery," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 8, pp. 10275–10286, Aug. 2022, doi: 10.1109/TITS.2021.3089675.

[24] S. Hulagu and H. B. Celikoglu, "Environment-friendly school bus routing problem with heterogeneous fleet: A large-scale real case," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 4, pp. 3461–3471, Apr. 2022, doi: 10.1109/TITS.2020.3036696.

[25] S. Hulagu and H. B. Celikoglu, "A multiple objective formulation of an electric vehicle routing problem for shuttle bus fleet at a university campus," in *Proc. 6th Int. Conf. Models Technol. Intell. Transp. Syst. (MT-ITS)*, Jun. 2019, pp. 1–5.

[26] S. Hulagu and H. B. Celikoglu, "An electric vehicle routing problem with intermediate nodes for shuttle fleets," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1223–1235, Feb. 2022.

[27] G. Ioannou, M. Kritikos, and G. Prastacos, "A greedy look-ahead heuristic for the vehicle routing problem with time windows," *J. Oper. Res. Soc.*, vol. 52, no. 5, pp. 523–537, May 2001.

[28] H. Y. Jeong, B. D. Song, and S. Lee, "Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones," *Int. J. Prod. Econ.*, vol. 214, pp. 220–233, Aug. 2019.

[29] H. Y. Jeong, B. D. Song, and S. Lee, "The flying warehouse delivery system: A quantitative approach for the optimal operation policy of airborne fulfillment center," *IEEE Trans. Intell. Transp. Syst.*, vol. 22, no. 12, pp. 7521–7530, Dec. 2021.

[30] H. Y. Jeong, B. D. Song, and S. Lee, "Optimal scheduling and quantitative analysis for multi-flying warehouse scheduling problem: Amazon airborne fulfillment center," *Transp. Res. C, Emerg. Technol.*, vol. 143, Oct. 2022, Art. no. 103831.

[31] P. Kabcome and T. Mouktonglang, "Vehicle routing problem for multiple product types, compartments, and trips with soft time windows," *Int. J. Math. Math. Sci.*, vol. 2015, no. 1, 2015, Art. no. 126754.

[32] G. Kim, Y.-S. Ong, C. K. Heng, P. S. Tan, and N. A. Zhang, "City vehicle routing problem (City VRP): A review," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 4, pp. 1654–1666, Aug. 2015.

[33] I. Koohi and V. Z. Groza, "Optimizing particle swarm optimization algorithm," in *Proc. IEEE 27th Can. Conf. Electr. Comput. Eng. (CCECE)*, May 2014, pp. 1–5.

[34] A. Lagorio, R. Pinto, and R. Golini, "Research in urban logistics: A systematic literature review," *Int. J. Phys. Distribution Logistics Manage.*, vol. 46, no. 10, pp. 908–931, Nov. 2016.

[35] A. Lambora, K. Gupta, and K. Chopra, "Genetic algorithm–A literature review," in *Proc. Int. Conf. Mach. Learn., Big Data, Cloud Parallel Comput. (COMITCon)*, Feb. 2019, pp. 380–384.

[36] M. Lee, J. Hong, T. Cheong, and H. Lee, "Flexible delivery routing for elastic logistics: A model and an algorithm," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 6864–6882, Jul. 2022, doi: 10.1109/TITS.2021.3063195.

[37] K. Leng and S. Li, "Distribution path optimization for intelligent logistics vehicles of urban rail transportation using VRP optimization model," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 1661–1669, Feb. 2022.

[38] K. Li, T. Zhang, R. Wang, Y. Wang, Y. Han, and L. Wang, "Deep reinforcement learning for combinatorial optimization: Covering salesman problems," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13142–13155, Dec. 2022.

[39] Y. Li, F. Chu, C. Feng, C. Chu, and M. Zhou, "Integrated production inventory routing planning for intelligent food logistics systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 3, pp. 867–878, Mar. 2019, doi: 10.1109/TITS.2018.2835145.

[40] Z. Li, A. Shalaby, M. J. Roorda, and B. Mao, "Urban rail service design for collaborative passenger and freight transport," *Transp. Res. E, Logistics Transp. Rev.*, vol. 147, Mar. 2021, Art. no. 102205.

[41] C. Liang, Y. Wang, B. Lu, and Y. Jin, "Joint scheduling of vertical and horizontal transportation for underground container logistics in seaport terminals," *PLoS ONE*, vol. 19, no. 11, Nov. 2024, Art. no. e0311536.

[42] Q. Luo, G. Wu, A. Trivedi, F. Hong, L. Wang, and D. Srinivasan, "Multi-objective optimization algorithm with adaptive resource allocation for truck-drone collaborative delivery and pick-up services," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 9, pp. 9642–9657, Apr. 2023, doi: 10.1109/TITS.2023.3267103.

[43] M. Chenglin, M. Hai-jun, X. Yang, D. jingfeng, C. Ma, and A. Lihua, "Study on the development mode of urban underground logistics system," *Service Sci. Manage. Res.*, vol. 3, pp. 7–12, Dec. 2014.

[44] Y. Marinakis, M. Marinaki, and A. Migdalas, "Particle swarm optimization for the vehicle routing problem: A survey and a comparative analysis," in *Proc. Conf. Logistics, Optim. Transp. (LOT)*. Springer, 2018, pp. 1163–1196.

[45] *Infrastructure Technologies: Challenges and Solutions for Smart Mobility in Urban Areas*, McKinsey Company, Chicago, IL, USA, 2023. [Online]. Available: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/infrastructure-technologies-challenges-and-solutions-for-smart-mobility-in-urban-areas

[46] *Drone Delivery: More Lift Than You Think*, McKinsey & Company, Chicago, IL, USA, 2022. [Online]. Available: https://www.mckinsey.com/industries/automotive-and-assembly/our-insights/drone-delivery-more-lift-than-you-think

[47] P. Mo, Y. Yao, A. D'Ariano, and Z. Liu, "The vehicle routing problem with underground logistics: Formulation and algorithm," *Transp. Res. E, Logistics Transp. Rev.*, vol. 179, Nov. 2023, Art. no. 103286, doi: 10.1016/j.tre.2023.103286.

[48] News, U.T. (2021). *Urban Logistics: Challenges, Innovations, and Sustainable Practices*. [Online]. Available: https://urbantransportnews.com/urban-logistics-challenges-innovations-and-sustainable-practices/

[49] *How Shared Mobility Impacts the Global Urban Landscape*, Oliver Wyman, New York, NY, USA, 2023. [Online]. Available: https://www.oliverwyman.com

[50] *Global Freight Demand To 2030*, Organisation for Econ. Co-operation Develop., Paris, France, 2023.

[51] Q. Pei, Y. Q. Fan, and S. D. Tang, "Key technical issues of ultra-deep waste transfer nodes in underground environmental logistics," *Chin. J. Underground Space Eng.*, vol. 8, no. 2, pp. 428–433, 2012.

[52] B.-J. Pielage, "Underground freight transportation. A new development for automated freight transportation systems in The Netherlands," in *Proc. IEEE Intell. Transp. Syst. (ITSC)*, Aug. 2001, pp. 762–767.

[53] J. P. Rodrigue, *Key Issues in Urban Freight Transportation*, 5th ed., Evanston, IL, USA: Routledge, 2020. [Online]. Available: https://transportgeography.org/?page_id=5955

[54] J. H. Holland, *Adaptation in Natural and Artificial Systems*. Cambridge, MA, USA: MIT Press, 1976, p. 529.

[55] P. Schiewe and A. Schöbel, "Periodic timetabling with integrated routing: Toward applicable approaches," *Transp. Sci.*, vol. 54, no. 6, pp. 1714–1731, Nov. 2020.

[56] Seoul Metrop. Government. (2024). *Seoul Subway Line 3 Weekday Schedule*. [Online]. Available: https://data.seoul.go.kr

[57] R. S. Sutton, *Reinforcement Learning: An Introduction*. Bradford, U.K.: A Bradford Book, 2018.

[58] *E-commerce and Logistics: A Match Made in Heaven*, United Nations Conf. Trade Develop., Geneva, Switzerland, 2020.

[59] T. Vidal, "Hybrid genetic search for the CVRP: Open-source implementation and SWAP neighborhood," *Comput. Operations Res.*, vol. 140, Apr. 2022, Art. no. 105643.

[60] J. G. S. N. Visser, "The development of underground freight transport: An overview," *Tunnelling Underground Space Technol.*, vol. 80, pp. 123–127, Oct. 2018.

[61] Y. Wang, S. Peng, X. Zhou, M. Mahmoudi, and L. Zhen, "Green logistics location-routing problem with eco-packages," *Transp. Res. E, Logistics Transp. Rev.*, vol. 143, Nov. 2020, Art. no. 102118.

[62] Y. Wang, X. Wang, Y. Wei, Y. Sun, J. Fan, and H. Wang, "Two-echelon multi-depot multi-period location-routing problem with pickup and delivery," *Comput. Ind. Eng.*, vol. 182, Aug. 2023, Art. no. 109385.

[63] Y. Wang, Z. Wei, S. Luo, J. Zhou, and L. Zhen, "Collaboration and resource sharing in the multidepot time-dependent vehicle routing problem with time windows," *Transp. Res. E, Logistics Transp. Rev.*, vol. 192, Dec. 2024, Art. no. 103798.

[64] Y. Wang, S. Luo, J. Fan, and L. Zhen, "The multidepot vehicle routing problem with intelligent recycling prices and transportation resource sharing," *Transp. Res. E, Logistics Transp. Rev.*, vol. 185, May 2024, Art. no. 103503.

[65] C. J. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, May 1992.

[66] L. Wei, Y. Chen, D. Guo, J. Ji, Z. Chen, and C. Zhuo, "A last-mile delivery system for underground logistics with 'self-pickup+' and 'home-entry+' modes," *Tunnelling Underground Space Technol.*, vol. 147, May 2024, Art. no. 105678.

[67] W. Wong and C. I. Ming, "A review on Metaheuristic algorithms: Recent trends, benchmarking and applications," in *Proc. 7th Int. Conf. Smart Comput. Commun. (ICSCC)*, Jun. 2019, pp. 1–5.

[68] Y. Xiaobin, C. Zhilong, and G. Dongjun, "Comprehensive evaluation of environmental and economic benefits of China's urban underground transportation construction projects," *J. Environ. Biol.*, vol. 36, no. 4, p. 733, Jul. 2015.

[69] Y. Xu, J. Dong, R. Ren, K. Yang, and Z. Chen, "The impact of metro-based underground logistics system on city logistics performance under COVID-19 epidemic: A case study of Wuhan, China," *Transp. Policy*, vol. 116, pp. 81–95, Feb. 2022.

[70] V. F. Yu, A. A. N. P. Redi, Y. A. Hidayat, and O. J. Wibowo, "A simulated annealing heuristic for the hybrid vehicle routing problem," *Appl. Soft Comput.*, vol. 53, pp. 119–132, Apr. 2017.

[71] M. Zhang, Y. Zeng, K. Wang, Y. Li, Q. Wu, and M. Xu, "Learned unmanned vehicle scheduling for large-scale urban logistics," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 7933–7944, Jul. 2024.

[72] C. Zheng, Y. Gu, J. Shen, and M. Du, "Urban logistics delivery route planning based on a single metro line," *IEEE Access*, vol. 9, pp. 50819–50830, 2021.

[73] T. Zhou and A. L. Zhou, "Optimization model of distribution route in underground logistics system based on the time-cost," *Logistics Eng. Manage.*, vol. 38, no. 8, pp. 60–62, 2016.