

Document Version

Final published version

Licence

CC BY

Citation (APA)

Stokkink, P., & Tran-Thi, A. P. (2026). A rolling horizon approach for multi-objective optimization of two-way logistics with postponed orders. *Computers and Industrial Engineering*, 217, Article 112046.
<https://doi.org/10.1016/j.cie.2026.112046>

Important note

To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Copyright

In case the licence states "Dutch Copyright Act (Article 25fa)", this publication was made available Green Open Access via the TU Delft Institutional Repository pursuant to Dutch Copyright Act (Article 25fa, the Taverne amendment). This provision does not affect copyright ownership.
Unless copyright is transferred by contract or statute, it remains with the copyright holder.

Sharing and reuse

Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy

Please contact us and provide details if you believe this document breaches copyrights.
We will remove access to the work immediately and investigate your claim.



A rolling horizon approach for multi-objective optimization of two-way logistics with postponed orders

Puong Anh Thi Tran ^a , Patrick Stokkink ^b ,* 

^a Faculty of Civil Engineering and Geosciences, Delft University of Technology, The Netherlands

^b Faculty of Technology, Policy, Management, Delft University of Technology, The Netherlands

ARTICLE INFO

Keywords:

Last-mile delivery

Two-way logistics

Multi-objective optimization

Rolling-horizon planning

ABSTRACT

Last-mile delivery of e-commerce is a complex mix of simultaneous delivery of goods and pickup of returned goods. With return rates being substantial in the e-commerce sector, this problem is highly relevant. To address this challenge, this study formulates a rolling horizon simultaneous pickup and delivery problem with time windows in a two-way logistics system. The proposed approach integrates a multi-objective mixed-integer linear programming model with a rolling-horizon framework, allowing forward deliveries and reverse pickups to be postponed based on different priorities and penalties, while capturing the interdependence between outbound and return flows. Economic efficiency, environmental impact, and social factors are explicitly incorporated into the objective function. For large-scale instances, an adaptive large neighborhood search with simulated annealing metaheuristic is designed and compared against a weighted-sum mixed integer linear program approach. Numerical experiments on Solomon datasets demonstrate that the proposed method efficiently achieves high-quality solutions, reveals trade-offs among sustainability objectives, and shows how coordinating forward and reverse flows can enhance operational adaptability. The results also highlight an unintended outcome of the proposed rolling-horizon planning: generation and propagation of backlog of demand. This highlights the need for adaptive mechanisms that can manage such issues in multi-period planning.

1. Introduction

The continued expansion of urbanization and e-commerce has intensified the challenges of city logistics, particularly in the Last-Mile Delivery (LMD) stage, involving transporting products from a warehouse or distribution hubs to the final destination of end consumers (MANIATIS, 2023). Given that LMD can represent approximately between 13% and 37% of total supply chain costs (Demir et al., 2022), poorly managed LMD operations arising from increased delivery frequency and order fragmentation can elevate operational expenses and carbon emissions for logistics service providers. Moreover, there is a growing volume of returned online purchases due to profit-oriented behavior, impulsive purchasing, product defects, and information asymmetry, which lead to further challenges for reverse logistics operations, requiring effective strategies to address these complexities (de Leeuw et al., 2016). Compared to traditional retail channels, e-commerce platforms tend to generate a higher percentage of product returns (Zhang et al., 2021). Considering the characteristics of returned items, an effective approach is to develop an integrated logistics system that simultaneously handles deliveries and pickups (Wang et al., 2015). Under such circumstances,

the joint management of forward logistics (delivering products to customers) and reverse logistics (collecting returned products back to suppliers' depots) enables significantly higher vehicle utilization with more effective and balanced routing in terms of freight flows. Thus, the optimal solution of the delivery–pickup logistics system also contributes to the sustainable development, as this challenge involves not only the economic impacts arising from transport costs but also with environmental matters and social dimensions of customer satisfaction (Zhang et al., 2021).

From an operations research perspective, these challenges are commonly modeled within the framework of the Vehicle Routing Problem (VRP), which focuses on designing efficient transportation routes to meet the demands of spatially distributed customers (Nepomuceno et al., 2023). Extensive research has been conducted with optimization approaches regarding vehicle routing, in which several different variations are largely driven by the diversity of real-world challenges in enhancing the efficiency of transportation systems (Braekers et al., 2016). Among these variants, the Vehicle Routing Problem with Simultaneous Pickup and Delivery (VRPSPD), originally proposed by

* Corresponding author.

E-mail address: p.s.a.stokkink@tudelft.nl (P. Stokkink).

Min (1989), specifically considers the forward distribution and reverse collection operations together with the aim of minimizing transportation costs and enhance overall system performance.

In this study, we address a multi-objective Simultaneous Pickup and Delivery Problem with Time Windows (SPDPTW) through a rolling-horizon framework for a two-way logistics system, where forward deliveries and reverse pickups are jointly optimized and can be postponed to the next day according to different priorities and penalty costs. This problem captures the dual-flow nature of modern e-commerce distribution, in which vehicles must coordinate outbound deliveries and inbound returns efficiently. We design an integrated optimization framework combining a multi-objective Mixed-Integer Linear Programming (MILP) model with the weighted-sum method for benchmarking and an Adaptive Large Neighborhood Search with Simulated Annealing (ALNS-SA) metaheuristic for solving large-scale instances. These solution approaches are embedded within a rolling horizon framework that allows for the incorporation of previously unfulfilled demand in next-day deliveries. Numerical experiments are performed using Solomon datasets to evaluate the effectiveness of the developed approach. The optimal solution not only highlights the trade-offs among the three sustainability pillars but also reveals how forward and reverse flows can complement each other to achieve sustainable operations.

The remainder of this paper is organized as follows. Section 2 provides a brief review of the relevant literature. In Section 3, we provide the problem description and present the mathematical formulation. Section 4 details the proposed methodology approach, while Section 5 indicates the results of the computational experiments. Finally, Section 6 concludes the paper with managerial insights and gives suggestions for future research.

2. Literature review

Overall, last-mile delivery primarily concerns the transportation of items from a central depot to designated urban destinations while overall costs are minimized (Cardenas et al., 2017), which is the fundamental focus of VRP (Toth & Vigo, 2014). A large number of studies have also explored various VRP variants in forward logistics, such as time-dependent VRP (Çimen & Soysal, 2017), VRP with pickup and delivery (Soysal et al., 2020), inventory routing problem (Soysal et al., 2021), pollution routing problem (Bektaş & Laporte, 2011), sustainable VRP (Dündar et al., 2021) and green VRP (Erdoğan & Miller-Hooks, 2012). While the VRPTW problem has been a common research focus in operations research for over five decades, a more recent development involves synchronization constraints, where multiple vehicles may be required to jointly serve a single customer. A detailed review of these new constraints in VRP can be found in Drexel (2012).

Moreover, due to the emerging trend of returns, exchanges, and damaged goods, an increasing interest is observed in extending VRP formulations to effectively manage reverse logistics operations (Iassinovskaia et al., 2017; Polat et al., 2015; Ramirez, 2012). Among these extensions, the Vehicle Routing Problem with Simultaneous Pickup and Delivery and Time Windows (VRPSPDTW), as defined by Angelelli and Mansini (2002), considers servicing each customer once within their designated time window while simultaneously handling both deliveries and pickups. This variant represents a particularly challenging combinatorial optimization problem, as it increases the complexity of the classic VRP with additional complex operational constraints (Wang et al., 2015).

However, of existing studies on the vehicle routing problem in reverse logistics, few have addressed the e-commerce sector, in spite of the rapid expansion of online retailing and the corresponding surge in product returns. There are significant distinctions between how e-commerce and traditional business models operate regarding the VRPRL. In conventional settings, product distribution involves transporting goods from warehouses to physical retail outlets. In contrast, e-commerce operators are responsible for arranging the direct delivery

of products to customers' locations or designated pickup points, which necessitates more complex and costly last-mile delivery operations compared with traditional distribution channels (Mallari et al., 2023). In particular, e-commerce logistics operations are characterized by larger overall volumes composed of smaller, more fragmented orders that must be collected, sorted, packaged, and distributed (Zhang et al., 2021). At the same time, the frequency of returns, exchanges, and damaged products also continues to increase, posing a significant challenge in managing extensive reverse logistics operations (de Leeuw et al., 2016). Several scholars have proposed similar definitions of reverse logistics (Kumar et al., 2017). Within the e-commerce context, returned products typically involve items which are purchased online but failed to meet customer expectations (Kokkinaki et al., 2001). Promotional campaigns also trigger impulsive purchasing with low-priced products that may not satisfy the actual needs of consumers. Another major reason of high return rates is the discrepancy between the physical items and their online representation. Several studies indicate that the high rate of returns also increase processing costs significantly for online retailers (Grewal et al., 2004; Griffis et al., 2012; Zhang, 2017). These findings suggest that the average costs of handling returns per product are generally higher in e-commerce than in traditional retail, primarily because conventional retailers can leverage their physical locations as additional channels for managing returned orders.

A growing number of studies in operations management have investigated reverse product flows and related warehousing activities (Alshamrani et al., 2007; Govindan et al., 2015; Ramanathan, 2011; Zhao et al., 2017). In practice, several companies have integrated and coordinated forward and reverse flows within their supply chains, thus establishing a 'closed-loop supply chain system' (Hassini et al., 2012). Among the limited studies addressing the VRPRL within e-commerce, Deng et al. (2014) achieved simultaneous minimization of forward and reverse logistics costs while maximizing customer satisfaction. Zhang et al. (2018) developed a bi-objective vehicle routing model to identify optimal solutions that simultaneously minimize transportation costs and load-dependent risks in service operations. Moons et al. (2019) proposed a VRP model that integrates order picking at distribution centers with the delivery of orders to final destinations in a B2C e-commerce setting, with the objective of minimizing total costs for warehouse and distribution operations while maintaining the desired service level. Similarly, Qiu et al. (2018) formulated an MILP model for solving production routing problems within the context of reverse logistics and remanufacturing. Kuo et al. (2016) investigated the VRP faced by local service companies, aiming to maximize the total quantity of served customers while minimizing waiting times by efficiently scheduling orders within the ongoing time period. Zhang et al. (2021) developed a MINLP model to primarily focus on minimizing total transportation cost and penalties for late deliveries within the ongoing time period instead of leaving unmet demand to be resolved in the next period. Li et al. (2021) extended previous research by monetizing and integrating customer satisfaction levels with financial factors into a single objective function. Although Belbağ (2025) accounted for the postponement of the pickup flow, there is neither explicit prioritization of forward demand over reverse nor realism of the forecast of return requests based on forward delivery, which makes e-commerce operations less adaptive.

Besides, the literature review made by Arevalo-Ascanio et al. (2024) and Tadaros and Migdalas (2022) both reveal that most of the existing studies related to last-mile delivery focus primarily on the economic dimension when defining modeling objectives. Some studies consider fuel consumption and emissions in LMD of e-commerce over recent years. Tiwari and Sharma (2023) investigated the effect of emissions on routing decisions by considering emission cost with a side constraint. Total emission cost should not be over the pre-determined budget at the end of the distribution process. Yu et al. (2024) considered a multi-objective LMD problem, aiming to minimize transportation costs and carbon emissions while maximizing customer satisfaction. The results

Table 1
Overview of relevant literature.

Ref	Problem	Model	Objective			Flow characteristics					Planning horizon
			Eco	Env	Soc	Forward	Reverse	Demand priority	Dynamic demand	Postponement	
Li et al. (2013)	LIRP	ILP	✓			✓	✓				Single
Ge et al. (2018)	2E-VRP	NLIP	✓			✓			✓		Multi-period
Heshmati et al. (2019)	GVRP	MILP	✓	✓		✓					Single
Moons et al. (2019)	I-OP-VRP	MILP	✓			✓					Single
Liu (2020)	PDP	ILP	✓			✓					Single
Li et al. (2021)	LRP	MO-IP	✓			✓	✓				Single
Zhang et al. (2021)	MVRPSPDTW	MILP	✓			✓	✓				Multi-period
Tao et al. (2022)	MD-CVRP-OSA	IP	✓			✓					Single
Wang et al. (2022)	LRP-PS	MILP	✓			✓					Single
Tiwari and Sharma (2023)	GVRP	MILP	✓			✓					Single
Kahalimoghadam et al. (2024)	CMDGVRP	MOP	✓	✓		✓					Single
Xiao et al. (2024)	GVRPD-SR	MILP		✓		✓					Single
Yu et al. (2024)	GRVP	MILP	✓	✓	✓	✓					Single
Belbağ (2025)	GvRP	MILP	✓	✓		✓	✓			✓	Single
This study	SPDPTW	MILP	✓	✓	✓	✓	✓	✓	✓	✓	Rolling

Note: LIRP: Location-inventory-routing problem, 2E-VRP: Two echelon vehicle routing problem, GVRP: Green vehicle routing problem, I-OP VRP: Integrated order picking-vehicle routing problem, PDP: Pickup and delivery problem, LRP: Location routing problem, MVRPSPDTW: Vehicle routing problem with simultaneous pickup and delivery with time windows, MD-CVRP-OSA: Multi-depot capacitated vehicle routing problem with order split and allocation, LRP-PS: Location-routing problem with pick-up stations, CMDGVRP: Collaborative multi-depot green vehicle routing problem, MOP: Multi-objective programming, ILP: Integer linear programming, IP: Integer programming, MILP: Mix integer linear programming, MO-IP: Multi-objective integer programming, NLIP: Non-linear integer programming.

indicate that the proposed algorithm decreases transportation costs and the amount of carbon emissions, along with higher service levels by improving customer satisfaction.

An overview of relevant literature on last-mile delivery within e-commerce context is given in Table 1. The review of the literature reveals that, to the best of our knowledge, there is currently no research considering the postponement of both forward deliveries and reverse pickups based on different priorities and penalty costs alongside the interdependency of those requests, according to the multiple objectives of sustainability in last-mile delivery. Moreover, the integration of a rolling-horizon planning framework into the SPDPTW context remains largely unexplored, despite its potential to better reflect the dynamic and uncertainty patterns of e-commerce operations. These unexplored aspects allow for a more robust and realistic operational strategy when forecasting and planning two-way logistics. Therefore, this study aims to fill this research gap by formulating a comprehensive multi-objective framework for the Simultaneous Pickup and Delivery Problem with Time Windows (SPDPTW) in last-mile e-commerce operations, which (i) explicitly incorporates the postponement of forward deliveries and reverse pickups, (ii) captures the interdependencies between delivery and return flows based on their priorities and penalty costs, and (iii) endogenously generates return demand within a rolling-horizon approach. By integrating economic, environmental, and social objectives, the model provides a realistic and sustainable planning strategy that enhances vehicle utilization, reduces operational costs, mitigates carbon emissions, and improves customer satisfaction. An adaptive Large Neighborhood Search with Simulated Annealing (ALNS-SA) algorithm is employed for its strong capability to explore solution spaces under tight constraints and its flexibility to integrate other effective heuristics, making it both efficient and adaptable for solving large-scale, complex e-commerce last-mile logistics problems.

3. Problem formulations

This section provides a description of the problem and its corresponding mathematical formulation.

3.1. Problem description

We address a vehicle routing problem with simultaneous pickup and delivery under time windows, incorporating the triple bottom line framework of sustainability with three dimensions, namely economic,

environmental, and social. A set of customers requires the delivery and the collection of products within specific time windows. Vehicles, starting and ending at a central hybrid depot, must operate within capacity and maximum route duration, while potentially serving multiple customers in a single route. Service can begin only after the earliest time window of customers, but vehicles may arrive earlier and wait. Due to limited vehicle capacity, it is possible that not all delivery and pickup demands are met in a single period; unmet quantities are carried over to subsequent periods with penalty costs incurred. Furthermore, the quantity of returns from each customer is dependent on previously fulfilled delivery demand. Besides, customers may be visited by more than one vehicle within the same period, reflecting practical e-commerce operations where orders are often fragmented and distributed across vehicles due to capacity and resource limitations. Imposing a single-visit constraint would reduce routing flexibility, potentially increasing unmet demand or fleet size, while also leading to higher computational complexity. The problem is to determine the optimal routing scheme for all vehicles that minimizes total operational costs, reduces greenhouse gas emissions, and optimizes drivers' working hours, thereby balancing the trade-offs among the three sustainability pillars. In addition, the following modeling assumptions are made:

- Each customer has a predefined time window; early arrival is permitted, but service starts only at the opening time
- Customers can be served by multiple vehicles within the same period for both delivery and pickup
- Unmet delivery or pickup quantities are postponed to the next period, incurring penalty costs

Fig. 1 illustrates the procedure of rolling horizon operation dealing with postponed requests across multiple periods. In every time interval, a static optimization problem is solved. The time intervals are connected in two ways: (1) Unmet forward and reverse demand are postponed until the next time period, (2) A percentage of served forward demand will be returned by the users in the next time interval.

3.2. Model formulation

Based on the problem described in the preceding section, an MILP model is formulated, and the sets, parameters, and decision variables are reported in Table 2.

The mathematical formulation then reads as follows. For the sake of readability, the constraints have been split up into subgroups.

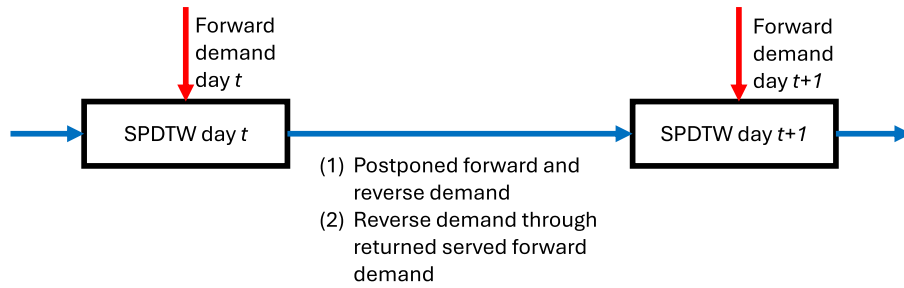


Fig. 1. Rolling horizon operation with exogenous demand flows marked in red and endogenous demand flows marked in blue.

Table 2

Notation.

Sets and indices		
N	Set of all nodes $i, j \in N$, including 0 and $n + 1$ as depot start and end	
C	Set of customer nodes $i, j \in C$ without depot	
V	Set of vehicles $v \in V$	
Parameters		
t_{ij}	Travel distance between node i and $j, i, j \in N, j \neq i$	
c_{ij}	Transportation cost per unit distance, $i, j \in N, j \neq i$	
d_i^c	Delivery demand of customer i in the current period, $i \in C$	
F_{iv}^p	Delivery quantity of customer i served by vehicle v in the previous period, $i \in C$	
U_i^d	Non-satisfied delivery demand of customer i in the previous period, $i \in C$	
U_i^r	Non-satisfied return demand of customer i in the previous period, $i \in C$	
δ_i	Return rate of customer $i, i \in C$	
λ_v	Maximum capacity of vehicle $v, v \in V$	
p_i	Penalty cost per unit of non-satisfied forward demand of customer	
θ	Importance weight of forward demand relative to return demand, where $0 \leq \theta \leq 1$	
m_v	Maximum allowed transportation duration of each vehicle	
s_i	Service time at customer node $i, i \in C$	
a_i	Earliest begin time of service at customer node $i, i \in C$	
b_i	Latest begin time of service at customer node $i, i \in C$	
c_v	Fixed cost of operating vehicle $v, v \in V$	
h_v	Hourly wage of driver	
e_v	Carbon emission of vehicle $v, v \in V$	
c_f	Fuel cost per liter	
f_v	Fuel consumption rate of vehicle $v, v \in V$	
γ	Social cost of carbon (\$/kg CO ₂)	
M	Large number	
w_1	Weight of economic objective	
w_2	Weight of environmental objective	
w_3	Weight of social objective	
Variables		
x_{ijv}	Binary variable, 1 if route from node i to node j served by vehicle v , 0 otherwise	{0,1}
z_{iv}	Binary variable, 1 if node i is visited by vehicle v , 0 otherwise	{0,1}
y_v	Binary variable, 1 if vehicle v is used, 0 otherwise	{0,1}
L_{0v}	Load of vehicle v when leaving the depot	
L_{iv}	Load of vehicle v remained after serving customer i	
α_{iv}	Service beginning time at customer i by vehicle v	
β_v	Time that vehicle v finishes all services	
F_{iv}^c	Delivery quantity of customer i served by vehicle v - forward flow	
R_{iv}^c	Return quantity of customer i served by vehicle v - return flow	
Q_i^d	Non-satisfied delivery quantity of customer i	
Q_i^r	Non-satisfied return quantity of customer i	

Objective:

$$\text{Min } Z = w_1 Z_{eco} + w_2 Z_{env} + w_3 Z_{soc} \quad (1)$$

$$Z_{eco} = \sum_{v \in V} (h_v \beta_v + c_v y_v) + \sum_{i \in N} \sum_{j \in N} \sum_{v \in V} t_{ij} x_{ijv} (c_f f_v + c_{ij}) + p_i \sum_{i \in C} (\theta Q_i^d + (1 - \theta) Q_i^r) \quad (2)$$

$$Z_{env} = \sum_{i \in N} \sum_{j \in N} \sum_{v \in V} e_v t_{ij} x_{ijv} \gamma \quad (3)$$

$$Z_{soc} = \sum_{v \in V} h_v \beta_v \quad (4)$$

Objective function (1) is a minimization of the weighted combination of economic, environmental, and social costs, capturing the trade-offs

among the three pillars of sustainability. The three objectives are given separately in Eqs. (2), (3), and (4). Eq. (2) calculates the economic cost, considering five components: driver wages, vehicle operating costs, fuel costs, transport costs, and penalty costs for non-satisfied forward and reverse demand. Eq. (3) computes the total carbon emissions, based on the specific emission factor of each vehicle. Eq. (4) considers the total working hours across all drivers.

Routing constraints:

$$\sum_{i \in N} x_{0iv} = \sum_{i \in N} x_{i0v} \quad \forall v \in V \quad (5)$$

$$\sum_{i \in C} x_{0iv} = y_v \leq 1 \quad \forall v \in V \quad (6)$$

$$\sum_{j \in C} x_{ijv} = \sum_{j \in C} x_{jiv} \quad \forall v \in V, \forall i \in C \quad (7)$$

$$x_{iiv} = 0 \quad \forall v \in V, \forall i \in C \quad (8)$$

$$\sum_{j \in N} x_{ijv} = z_{iv} \quad \forall v \in V, \forall i \in N \quad (9)$$

$$\sum_{v \in V} z_{iv} \geq 1 \quad \forall i \in C \quad (10)$$

Constraints (5) guarantee that every vehicle begins and finishes its route in the depot. Constraints (6) emphasize that each vehicle can be used or not. Constraints (7) specify that if vehicle v arrives at node i , then the same vehicle must leave the node i . Constraints (8) ensure that a vehicle v does not travel directly from a customer node i back to the same node i . A self-loop is only allowed at the depot to account for an unused vehicle. Constraints (9) and (10) together maintain that each destination node is visited from nodes that have already been visited and each customer node can be visited by multiple vehicles.

Forward and reverse demand service constraints:

$$\sum_{v \in V} F_{iv}^c + Q_i^d \geq d_i^c + U_i^d \quad \forall i \in C \quad (11)$$

$$F_{iv}^c \leq (d_i^c + U_i^d)z_{iv} \quad \forall i \in C \quad (12)$$

$$\sum_{v \in V} R_{iv}^c + Q_i^r \geq \sum_{v \in V} F_{iv}^p \delta_i + U_i^r \quad \forall i \in C \quad (13)$$

$$R_{iv}^c \leq (\sum_{v \in V} F_{iv}^p \delta_i + U_i^r)z_{iv} \quad \forall i \in C \quad (14)$$

Constraints (11) and (12) ensure that forward demands of customers may not be fully satisfied in each period, and those are not served if the vehicle does not visit. In particular, the delivery demand of the current period includes both the newly generated demand and the unsatisfied quantity carried over from the previous period. Besides, Constraints (13) and (14) ensure that reverse demands of customers may not be fully satisfied in each period, and those are not served if the vehicle does not visit. The reverse flow demand depends not only on the return rate and the forward delivery in the previous period, but also includes any unsatisfied reverse demand from that period.

Load and capacity constraints:

$$L_{0v} \geq \sum_{i \in C} F_{iv}^c \quad \forall v \in V \quad (15)$$

$$L_{jv} \geq L_{iv} - F_{jv}^c + R_{jv} - M(1 - x_{ijv}) \quad \forall i, j \in C, j \neq i, \forall v \in V \quad (16)$$

$$L_{0v} \leq \lambda_v \quad \forall v \in V \quad (17)$$

$$L_{iv} \leq \lambda_v z_{iv} \quad \forall i \in C, \forall v \in V \quad (18)$$

The initial load of each vehicle corresponds to the total forward quantities it is assigned to serve, which is calculated as in Constraints (15). Constraints (16) ensure the load balance of vehicles after they finish serving each node along their route. Constraints (17) and (18) are the vehicle capacity constraints for initial and subsequent loads.

Time window and shift duration constraints:

$$\alpha_{jv} \geq \alpha_{iv} + s_i + t_{ij} - M(1 - x_{ijv}) \quad \forall i, j \in N, j \neq i, \forall v \in V \quad (19)$$

$$a_i z_{iv} \leq \alpha_{iv} \leq b_i + M(1 - z_{iv}) \quad \forall i \in N, \forall v \in V \quad (20)$$

$$\beta_v = \sum_{i \in N} \sum_{j \in C} t_{ij} x_{ijv} + \sum_{i \in C} \sum_{j \in C} s_i x_{ijv} \quad \forall v \in V \quad (21)$$

$$\beta_v \leq m_v \quad \forall v \in V \quad (22)$$

Constraints (19) ensure that if vehicle v is traveling from i to j , the service start time at j accounts for both the service duration at node i and the travel time between i and j . Otherwise, this constraint becomes non-binding due to the use of the large constant M . The time window constraints, as given in (20), ensure that the service start time at each node respects the earliest and latest allowable times defined by the time windows in case that node is visited by the vehicle. Constraints (21) and (22) together present the maximum allowed transportation duration limit for each vehicle.

Domain constraints:

$$x_{ijv}, z_{iv}, y_v \in \{0, 1\} \quad \forall i, j \in N, \forall v \in V \quad (23)$$

$$L_{0v}, L_{iv}, \alpha_{iv}, \beta_v, F_{iv}, R_{iv}, Q_i^d, Q_i^r \geq 0 \quad \forall i \in C, \forall v \in V \quad (24)$$

Constraints (23) and (24) define the variables as binary and continuous, respectively.

4. Solution approach

4.1. Framework of ALNS-SA algorithm

The Adaptive Large Neighborhood Search (ALNS) framework proposed by Ropke and Pisinger (2006) extends from the Large Neighborhood Search method of Shaw (1998), which has been widely applied to various vehicle routing problems. Its main principle is to iteratively improve solutions by employing multiple destruction and repair operators. This study adopts an ALNS approach, outlined in Algorithm 1, which is inspired by the work of Ma and Yang (2025). The method integrates a set of different destruction and repair operators into a Simulated Annealing (SA) framework to escape local optima and improve solution quality. Solutions satisfy all routing and demand satisfaction constraints by definition, while time window constraints and vehicle capacity are incorporated as soft penalties. Specifically, Constraints (18), (20) (right-hand side only), and (22) are relaxed and treated as soft penalties in case of violations. The ALNS framework then explicitly handles these violations, as explained in Section 4.7.

The ALNS algorithm begins with an initial solution (Line 2) generated by the nearest neighbor heuristic (Lawler, 1985). For each iteration, seven destruction operators remove a subset of customer nodes (Line 10), after which five repair operators reinsert them into the routes to form a new solution (Line 11). Each operator is assigned a score and weight, which are dynamically updated considering the performance of the new solution relative to the previous one (Lines 20–30). During the search process, operator selection is guided by these scores and weights. Finally, the SA criterion in Line 14 determines whether the new solution is accepted, with a controlled probability of accepting worse solutions to enhance diversification.

4.2. Initial solution process

The algorithm begins by constructing an initial feasible solution using the Nearest Neighbor Heuristic (NNH). Each vehicle starts from the depot and sequentially visits the nearest feasible customer while respecting vehicle capacity constraints. The algorithm proceeds iteratively until every customer has been assigned to a route.

4.3. Destroy operator

The destroy phase partially removes customers from the current solution to diversify the search and enable reconstruction of potentially better solutions. The following seven destroy operators are adopted, of which the first five are adapted from Ma and Yang (2025).

- The **Random Destroy Operator** aims to increase solution diversity and prevent premature convergence, especially for scattered customer distributions, by randomly selecting and removing q customers from existing routes.
- The **Similarity Destroy Operator** specifically deals with dense customer groups by removing a set of nodes having high spatio-temporal correlation to force path reconstruction and reduce redundant driving. It removes a total of q customers that are most similar to a randomly chosen customer i that is considered a candidate seed. The degree of correlation is determined based on following criterion:

$$R(i, j) = \frac{1}{c'_{ij} + V_{ij}}$$

Algorithm 1: ALNS-SA Metaheuristic

Data: Initial solution of vehicle routes R_{init} from nearest neighbor heuristic, maximum iterations I_{max} , initial temperature T_0 , minimum temperature T_{min} , cooling rate c

Result: Best heuristic solution (R_{best}, C_{best})

- 1 Initialize weight vectors w_d, w_r and score vectors s_d, s_r for all destruction and repair operators;
- 2 $R_{current} \leftarrow R_{init}$;
- 3 $R_{best} \leftarrow R_{init}$;
- 4 $C_{current} \leftarrow Evaluate(R_{current})$;
- 5 $C_{best} \leftarrow C_{current}$;
- 6 $T \leftarrow T_{init}$;
- 7 **while** $T > T_{min}$ **and** $iter < I_{max}$ **do**
- 8 Select destruction operator d using roulette-wheel(w_d);
- 9 Select repair operator r using roulette-wheel(w_r);
- 10 $R_{partial} \leftarrow$ apply destruction operator d to $R_{current}$;
- 11 $R_{new} \leftarrow$ apply repair operator r to $R_{partial}$;
- 12 $C_{new} \leftarrow Evaluate(R_{new})$;
- 13 $\Delta C \leftarrow C_{new} - C_{current}$;
- 14 **if** $\Delta C < 0$ **or** $rand() < \exp(-\Delta C/T)$ **then**
- 15 $R_{current} \leftarrow R_{new}$;
- 16 $C_{current} \leftarrow C_{new}$;
- 17 **if** $C_{new} < C_{best}$ **then**
- 18 $R_{best} \leftarrow R_{new}$;
- 19 $C_{best} \leftarrow C_{new}$;
- 20 Increase scores $s_d[d], s_r[r]$ by reward for improvement;
- 21 **else**
- 22 Increase scores $s_d[d], s_r[r]$ by reward for acceptance;
- 23 **end**
- 24 **else**
- 25 Penalize scores $s_d[d], s_r[r]$;
- 26 **end**
- 27 Update usage counters $\epsilon_d[d]$ and $\epsilon_r[r]$;
- 28 **if** $iteration \bmod update_interval = 0$ **then**
- 29 Update operator weights w_d, w_r based on scores and usage;
- 30 **end**
- 31 $T \leftarrow T \cdot c$
- 32 **end**
- 33 **return** R_{best}, C_{best}

in which c'_{ij} is the normalized Euclidean distance, ranging from [0,1], and V_{ij} indicates if they belong to the same route. This operator improves path integration efficiency for densely distributed customer groups.

- The **Maximum Saving Cost Destroy Operator** alleviates vehicle capacity shortages by removing customer nodes that offer the largest cost savings upon removal. For each customer i on a route, the saving cost C_i is calculated as $D(i-1, i) + D(i, i+1) - D(i-1, i+1)$, where $D(i, j)$ is the distance between customers i and j . The operator then removes the top q customers with the highest ΔC_i values. This strategy is particularly effective for scenarios where vehicle capacity is highly constrained.
- The **Destroying Vehicle Destroy Operator** optimizes resource allocation by identifying and removing high-cost vehicle routes, addressing low vehicle utilization which is often caused by time window conflicts. The comprehensive cost evaluation for each vehicle is computed using the formula as follows:

$$Z = D_{total} \cdot \tau + \frac{T_{wait}}{N_{cust}} \cdot \nu$$

where D_{total} is the total travel distance, T_{wait} is the total waiting time, N_{cust} is the number of visited customers per vehicle route, and τ, ν are the respective weighting coefficients. The vehicle associated with the largest cost is selected and removed, generating a new search space. In this operator, exactly one vehicle route is removed in each iteration to balance diversification and computational efficiency. This evaluation mechanism improves overall vehicle resource utilization more effectively than simply optimizing the path length.

- The **Maximum Waiting Time Destroy Operator** improves time feasibility by removing vehicle paths with excessively long waiting times, especially under strict time window constraints. In the search process, exactly one vehicle with the longest waiting time, including penalties, is selected for removal in each iteration. This strategy significantly reduces time window conflicts and improves time compliance.
- The **2-Opt Operator**, which was initially introduced by Croes in 1958 (Croes, 1958), is a simple local search algorithm and widely applied to the Traveling Salesman Problem (TSP). This method eliminates route crossings by reordering edges, thereby improving an initial feasible solution through successive 'inversion' until a local optimum is reached and no further improvement is possible.
- The **Shaw Removal Operator**, first introduced by Shaw (1997), aims to remove customers that are closely related, according to multiple aspects. This study uses three weighted criteria to evaluate this relatedness, namely capacity, distance, and time. Capacity compares the load requirements of each request, distance accounts for the path length between those two locations, and time measures the temporal proximity of visits. These components are normalized and combined with specific weights to compute the overall relatedness, as shown in the following formula:

$$R_{ij} = \mu_1 \frac{t_{ij}}{\max(t)} + \mu_2 \frac{|a_i - a_j| + |b_i - b_j|}{\max(|a_i - a_j| + |b_i - b_j|)} + \mu_3 \frac{|d_i - d_j|}{\max(|d_i - d_j|)}$$

where t_{ij} denotes the travel time or distance between customers i and j ; a_i, b_i are the start and end of the time window for customer i ; d_i represents the demand of customer i ; and μ_1, μ_2, μ_3 are weighting coefficients.

The operator begins by randomly selecting an initial request and iteratively removing other q requests that have the smallest relatedness values relative to the initial one. This process promotes focused yet diverse neighborhood exploration by targeting related customers, enabling substantial local improvements while preserving feasibility.

4.4. Repair operator

After destruction, the repair phase reinserts the removed customers into the partial solution while maintaining feasibility. The following five repair operators are adopted from Ma and Yang (2025):

- The **Global Optimal Repair Operator** aims to minimize the total travel distance by evaluating insertion positions across every route for all unassigned customers, making it appropriate for handling mixed distribution contexts. This globally optimal insertion strategy calculates the distance increase for all insertion candidates and then selects the position leading to the smallest increase. Besides, the operator simultaneously enforces all constraints regarding time window and vehicle loads. If there is no existing path satisfying those constraints to insert a new customer, the algorithm will create the new path, with its cost increment based on the round-trip distance between the warehouse and the customer.
- The **Minimum Insertion Cost Repair Operator** employs a greedy strategy to generate feasible solutions and improve convergence speed quickly. It functions by calculating the minimum

Table 3
Score updating rules.

Score	Condition
s_1	Operator score if new solution outperforms global optimum
s_2	Operator score if new solution improves current but not global optimum
s_3	Operator score if new solution equals current
s_4	Operator score if new solution is worse than current

distance increase for each customer to be inserted across all feasible positions across all routes. The customer who has the largest ‘minimum insertion cost’ is then identified and first inserted into its pre-calculated best position. The process repeats until all removed customers have been assigned to the routes. If no feasible location exists in the current routes, a new path is created.

- The **Random k -Repair Operator** is designed to balance exploration and exploitation by introducing random perturbation, thereby addressing the lack of diversity in single insertion strategies. It calculates the distance increase for each candidate position for insertion and randomly selects one from the top k positions associated with the smallest increases. The value of k is dynamically adjusted with the iteration count: $k = \lfloor \text{iterationnumber} / \omega \rfloor + \phi$, starting at ϕ and increasing by 1 every ω iterations.
- The **Regret Criterion Repair Operator** prioritizes the insertion of customer nodes with high regret values to minimize path length fluctuations. For each customer, it calculates the regret value R_i by accumulating the difference of cost increment between the most optimal position ($\Delta_{i,1}$) and the j th optimal position ($\Delta_{i,j}$) for insertion, using the formula:

$$R_i = \sum_{j=2}^K (\Delta_{i,j} - \Delta_{i,1})$$

The customer with the highest regret value is then inserted at its minimum regret position. This strategy effectively prevents the accumulation of costs caused by delaying the insertion of critical customers.

- The **Minimum Waiting Time Repair Operator** strictly controls time window constraints to prevent time conflict accumulation, enhancing time feasibility, particularly in scenarios with strict time windows. It employs a greedy insertion algorithm that considers the waiting time for each candidate insertion position and selects the minimum one. The waiting time is evaluated as the difference between the arrival time at a customer and the sum of the departure time from the preceding node and the travel time, in which the arrival time is defined as the maximum of either the departure time plus travel time or the earliest serving time in their time window.

4.5. Adaptive selection strategy

A key feature of this algorithm is the adaptive selection strategy, which enhances adaptability via the process of dynamically adjusting operator selection probabilities. This strategy integrates a roulette wheel mechanism presented in the study of [Ma and Yang \(2025\)](#), ensuring that all operators can be selected, while those with higher weights are more frequently chosen.

At the start, equal weights are assigned to all operators. Throughout the search process, these weights are iteratively updated according to the quality of the solutions obtained. Operators producing better solutions receive higher scores and selection probabilities, whereas less effective ones with lower scores are retained for occasional use to maintain search diversity (see [Table 3](#)).

The ALNS iteration process is divided into several segments, each of length l_s . At the end of each segment, the weight of operator i is updated according to the formula as below:

$$\omega_{new} = \alpha \omega_{cur} + (1 - \alpha) \frac{s_{cur}}{\epsilon_{cur}} \quad (25)$$

in which ω_{new} denotes the updated operator weight for the next segment, ω_{cur} denotes the current operator weight, s_{cur} denotes the current score, ϵ_{cur} denotes the latest frequency the operator has been used, and α is a control parameter.

4.6. Acceptance and stopping criteria

To help the algorithm escape local optima, the acceptance criterion based on the SA algorithm is incorporated into the proposed framework. In particular, a neighboring solution generated by the removal and insertion procedures is always accepted if it improves the current solution, otherwise it is accepted with a probability of $e^{-\frac{f(S_{curr}) - f(S_{new})}{T}}$, in which T denotes the temperature, S_{curr} is the current solution, and S_{new} is the neighboring solution. The initial temperature is set to a constant T_0 and gradually reduced by a factor c to $c \cdot T$ during the algorithm’s iterations. The value of this factor is between 0 and 1 and typically close to 1 to ensure slow cooling. The primary purpose of incorporating SA is to enhance search diversification by allowing worse solutions to be accepted early in the process.

The algorithm terminates when the number of iterations goes beyond a threshold ψ_{ite} or the temperature reaches the minimum value T_{min} .

4.7. Strategies for handling infeasible solutions

To account for infeasible solutions, the objective function is adjusted by incorporating penalty terms that penalize violations of key constraints, namely vehicle capacity and time windows. The updated objective function can be expressed as:

$$F(X) = f(X) + \eta_1 \sum_{i \in C} \max(0, t_i - b_i) + \eta_2 \sum_{v \in V} \max(0, \beta_v - m_v) + \eta_3 \sum_{v \in V} \max(0, \sum_{i \in C} L_{iv} - \lambda_v) \quad (26)$$

in which $f(X)$ is the original objective function, while $F(X)$ is the new objective function that incorporates penalty terms during the heuristic search. The three penalty terms are weighed by η_1 , η_2 , and η_3 . The first penalty component measures time window violations for customer i as the lateness of actual service start time t_i relative to the upper bound b_i . The lower bound a_i is still enforced as a hard constraint, assuming that vehicles wait when they arrive before the start time. The second penalty component computes the constraint violations regarding the maximum route time of each vehicle v as the excess of total transportation duration β_v over the limit m_v . The third penalty component computes the capacity violation of each vehicle v as the excess of total vehicle load L_{iv} over the maximum vehicle capacity λ_v . Total capacity at the depot remains enforced as a hard constraint. By integrating these penalty terms, the algorithm can explore infeasible solutions while gradually shifting the search toward feasibility.

5. Result and discussion

5.1. Experimental setting

The performance of the ALNS-SA algorithm was evaluated using experimental cases derived from the classical benchmark by [Solomon \(1987\)](#), which consists of 56 instances involving 100 customers. A return rate of 14% was reported as the average share of returned items purchased online in Europe in 2025 ([sta, 2020](#)), reflecting realistic e-commerce operations. We note that these numbers will typically deviate between product groups and customer types, but in the absence

Table 4
Settings of parameters.

Category	Parameter	Description	Value
MILP	c_{ij}	Transportation cost per unit distance	30
	δ	Return rate of customer	0.14
	λ	Maximum capacity of vehicle	60
	p_i	Penalty cost per unit of non-satisfied demand	50
	θ	Importance weight of forward relative to reverse demand	0.8
	m_v	Maximum allowed transportation duration of each vehicle	1000
	c_v	Fixed cost of operating vehicle	100
	h_v	Hourly wage of driver	15
	e_v	Carbon emission of vehicle	0.1
	c_f	Fuel cost per liter	0.5
	f_v	Fuel consumption rate of vehicle	0.1
	γ	Social cost of carbon	0.1
	M	Large number	10^6
	w_1	Weight of economic objective	1
	w_2	Weight of environmental objective	1
w_3	Weight of social objective	1	
ALNS-SA	q	Number of customers to be removed (Random Destroy)	15
	τ	Weight of total travel distance in cost evaluation (Destroying Vehicle)	0.7
	v	Weight of mean waiting time in cost evaluation (Destroying Vehicle)	0.3
	ϕ	Initial value of k insertion positions (Random k -repair)	5
	ω	Number of iterations to update k value (Random k -repair)	500
	α	Parameter for updating weights of operators	0.9
	ψ_{ite}	Maximum number of iterations	3000
	l_s	Length of each segment	50
	T_0	Initial temperature	2
	T_{\min}	Minimum temperature	0.01
	c	Cooling rate	0.9985
	η_1, η_2, η_3	Penalty terms in ALNS objective	$10^5, 10^5, 10^5$
	s_1	Operator score if new solution outperforms global optimum	6
	s_2	Operator score if new solution improves current but not global optimum	3
	s_3	Operator score if new solution equals current	1
s_4	Operator score if new solution is worse than current	0	

of more detailed data on returns, this number is used as the constant return rate through the experiments. For small-scale experiments of up to 25 customers, the ALNS-SA results were compared to exact solutions obtained using Gurobi solver to assess computational efficiency and solution quality. For large-scale experiments of 100 customers, ALNS-SA was applied due to the intractability of the exact model within reasonable time limits.

The ALNS-SA algorithm and Gurobi model were both implemented and executed in Python on a Windows 11 system, equipped with an Intel Core i7-11370H CPU (3.00 GHz) and 16 GB of RAM. The time limit for Gurobi was set to 1800 s, with optimality gaps reported. All parameter values were chosen based on practical experience and initial testing (see Table 4). The equal-weight configuration ($w_1 = w_2 = w_3 = 1$) is adopted as the baseline scenario, representing a neutral decision-making perspective in which economic, environmental, and social objectives are treated as equally important. This balanced configuration allows the model to provide solutions that reflect overall sustainability performance rather than prioritizing a single dimension.

5.2. Result and analysis

5.2.1. Comparison on small instances

Table 5 presents the comparison between Gurobi and ALNS-SA for problem sizes from 5 to 25 customers. The objective function represents the weighted sum of economic, environmental, and social objectives, while the constraint violations column indicates the number of customers whose any constraints, namely time windows, maximum route time, or vehicle capacity, are violated. For the exact formulation, such violations are not permitted by definition of the problem. For the ALNS approach, violations may occur against a penalty.

For the smallest instance (5 customers), both approaches achieve identical solutions in negligible computation time. However, for 10 customers and above, Gurobi hits the time limit with substantial optimality gaps. In contrast, ALNS-SA consistently produces solutions with better objective values, while significantly reducing computation times. This

result shows the strong efficiency advantage of metaheuristics, even for small and medium-sized instances.

5.2.2. Large-scale instance performance

For the large-scale instances with 100 customers, Gurobi is computationally impractical, while ALNS-SA obtains solutions within reasonable computation times. The algorithm effectively balances the three sustainability objectives and accommodates stochastic return demands, supporting its applicability to real-world urban logistics planning.

Seven Solomon benchmark instances (rc101–rc107) are used to represent customer demand for actual deliveries (forward demand) and product returns (reverse flow) over seven consecutive days in a week, providing insight into the dynamic interaction between forward and reverse demand under a rolling horizon strategy. Fig. 2 illustrates the evolution of the objective value, where the lower curve represents the best known solution and the upper curve shows the cost of accepted solutions across iterations. The typical pattern of a Simulated Annealing metaheuristic is observed with large fluctuations occurring in the early stages due to the high temperature, allowing even poor solutions to be accepted. As the temperature decreases, fluctuations diminish and the search gradually converges toward better solutions. The best solution stabilizes after approximately 1400 iterations, indicating effective convergence of the algorithm.

With an average computation time of 240 s per run, the proposed ALNS-SA approach demonstrates both practicality and robustness for addressing large-scale, multi-period urban logistics problems, particularly under realistic e-commerce return scenarios. Although the experiments were conducted on datasets of 100 customers, the ALNS-SA framework indicates strong potential for handling much larger problem instances. For instance, Ropke and Pisinger (2006) successfully employed ALNS to solve the pickup and delivery problem with time windows, maintaining efficiency with more than 500 customers.

The output of the optimal solution illustrated in Fig. 3 indicates that reverse demand remains relatively stable and well-served, with satisfaction rates between 60% and 85%, whereas forward demand

Table 5
Comparison of Gurobi and ALNS performance for small instances.

No. of cus.	Gurobi				ALNS			
	Obj.	Cons. violations	CPU (s)	Gap (%)	Obj.	Cons. violations	CPU (s)	Diff (%)
5	6661.25	0	3	0.00%	6661.25	0	5.1	0.00%
10	13172.87	0	1800	19.69%	13083.87	0	42.5	-0.68%
15	16242.51	0	1800	44.19%	16143.56	0	29.1	-0.61%
20	24022.79	0	1800	33.87%	23874.32	0	41.3	-0.62%
25	26326.52	0	1800	47.69%	25950.77	0	24.6	-1.43%

For Gurobi, the gap denotes the difference between the best integer solution and the lower-bound of the linear relaxation. For ALNS, the “Diff” denotes the difference between the objective obtained by the ALNS algorithm and the best integer solution obtained by Gurobi.

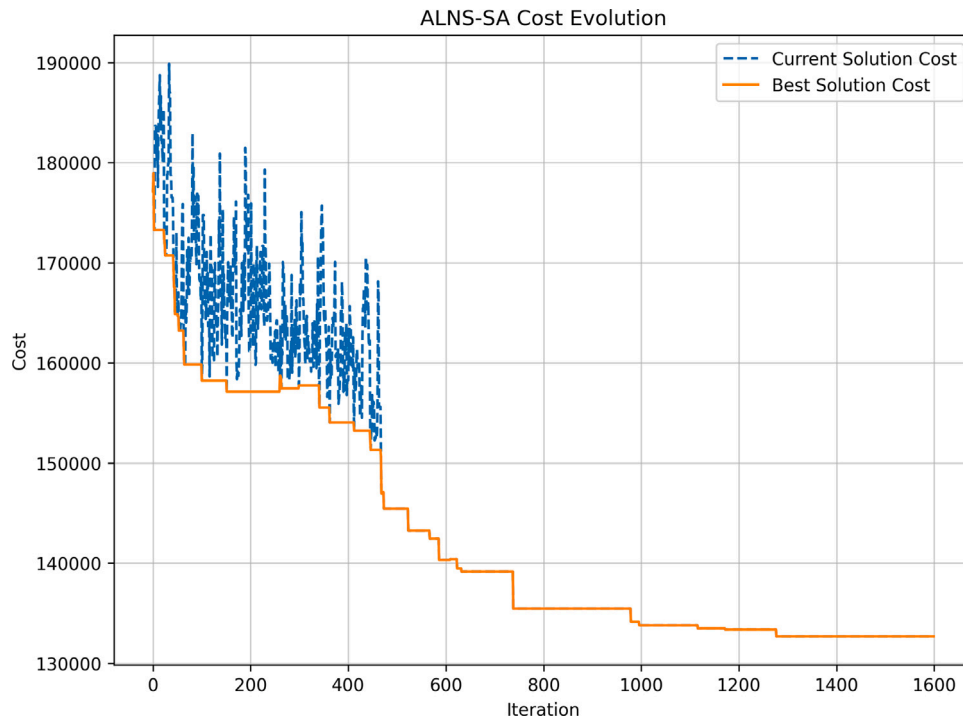


Fig. 2. ALNS-SA cost evolution.

satisfaction declines as cumulative unmet demand increases over time, showing that return pickups are generally easier to be handled within the available capacity. This reflects the inherent challenges posed by capacity constraints and carry-over effects in large-scale, multi-period VRPs, particularly when forward demand substantially exceeds return demand.

From a managerial perspective, the observed accumulation of unmet forward demand, as shown in Fig. 3, indicates that rolling-horizon planning can unintentionally generate backlog propagation when available fleet capacity is insufficient to absorb peak delivery demand. If left unmanaged, such backlog issues may lead to increasing service delays, higher penalty costs, and declining customer satisfaction. Under such circumstances, logistics managers can respond by combining adaptive metaheuristic optimization with proactive strategies such as implementing dynamic capacity policies, temporarily increasing fleet availability during peak demand periods, adjusting penalty structures to incentivize earlier service completion, or reallocating resources between delivery and pickup operations.

Moreover, these findings also highlight that reverse demands are typically easier to integrate into routing plans due to their lower volume and flexible timing requirements. Therefore, it could be suggested that logistics operators should prioritize the fulfillment of delivery demands when service reliability is critical, as reverse operations can be absorbed without causing major disruptions.

Fig. 4 illustrates the load dynamics of one specific vehicle on two representative days in a week, demonstrating the effectiveness of the algorithm in maintaining resource capacity while serving both forward and reverse flows. In both cases, the vehicle leaves the depot at or near full capacity, and its load gradually decreases as deliveries are made, while cumulative reverse quantities increase along the route as returns are collected. The curve of served forward demand by the vehicle for each visited customer highlights the adaptability of delivery quantities according to customer needs and capacity constraints. The patterns observed on both days are largely consistent, indicating stable routing scheme and demand conditions within the rolling horizon. Minor differences in unloading rates and collected return volumes arise from small daily fluctuations in customer demand and route allocation, particularly as multiple vehicles can jointly serve the same customer. Overall, the solution guarantees that the vehicle consistently operates within capacity limits, indicating the robustness of the algorithm in handling complex pickup and delivery scenarios over multiple planning horizons.

It can also be derived that allowing multiple vehicles to serve each customer contributes to this stability by enabling partial fulfillment strategies when capacity or time window limitations do not allow complete service by a single vehicle. This operational flexibility is particularly valuable in e-commerce industry, which is characterized by fragmented orders and uncertain demand realization.

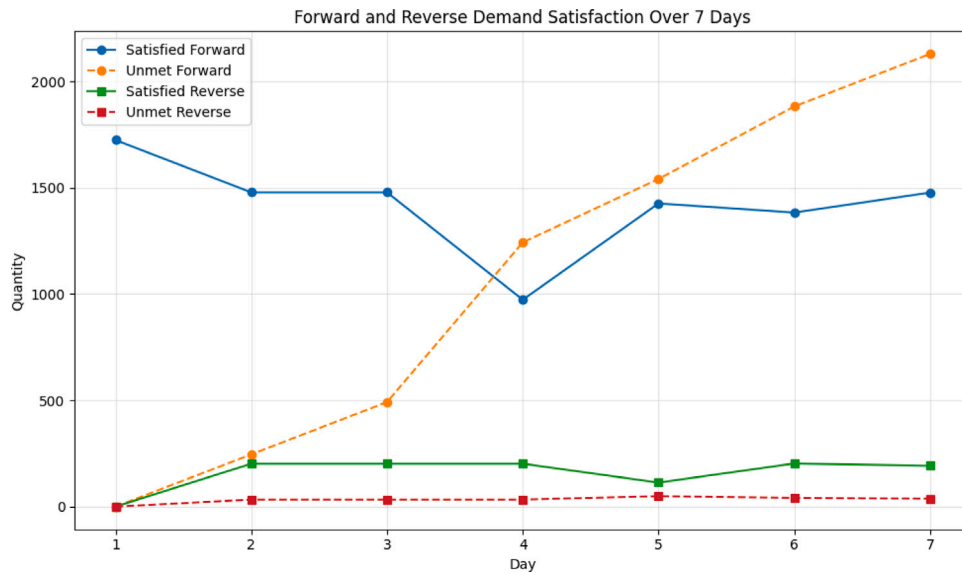


Fig. 3. Forward and reverse demand satisfaction over 7 days.

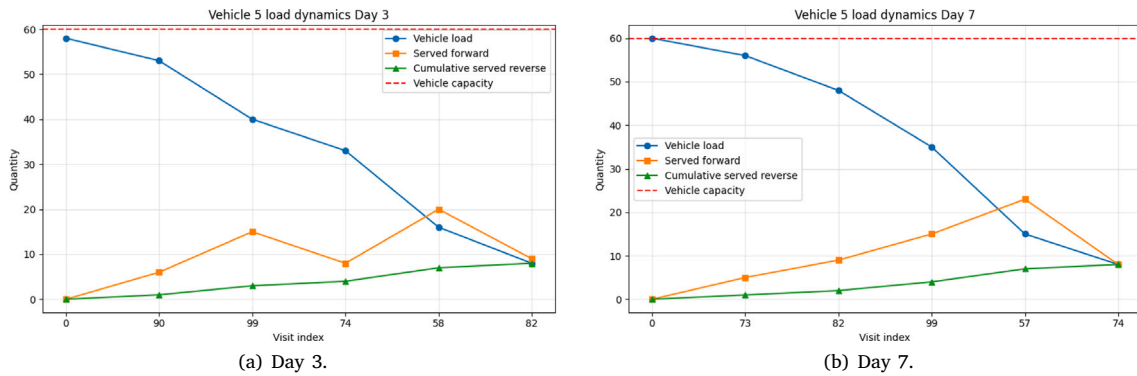


Fig. 4. Vehicle 5 load dynamics.

Table 6

Main results for different optimization scenarios.

Scenario	Economic	Environmental	Social
Economic only	100,083.06	14.00	30,984.69
Environmental only	103,136.80	12.68	41,702.39
Social only	162,885.60	41.26	15,107.20
Multi-objective (Equal weights)	107,159.85	16.98	32,204.29

Overall, it can be observed that the rolling-horizon approach successfully captures trade-offs between short-term efficiency and long-term service reliability in realistic operational scenarios. While postponement allows temporary relief of capacity pressure, excessive reliance on carry-over may lead to cumulative backlog effects. Logistics operators should therefore closely keep track of backlog indicators and trigger corrective adjustments once these levels exceed acceptable service thresholds.

5.3. Sensitivity analysis

A sensitivity analysis was conducted to examine how changes in the problem environment influence the proposed models, focusing on the objective weights, the penalty cost associated with unmet demand, and the modeling characteristics of carry-over flexibility.

5.3.1. Objective weights

To investigate how the model addresses real-world complexities and conflicting priorities among objectives, several weight configurations (w_1, w_2, w_3) was applied, ranging from single-objective emphasis (1, 0, 0), (0, 1, 0), (0, 0, 1) to balanced equal weights (1, 1, 1), as well as mixed priority sets with preselected values, such as (0.8, 0.1, 0.1), (0.3, 0.1, 0.6). The extreme cases are reported in Table 6 and the mixed priorities are given in Fig. 5. In general, the configuration of using equal weights serves as the baseline scenario because it represents a neutral sustainability policy and enables decision makers to observe inherent trade-offs without specifically imposing strategic preferences. This configuration is also particularly important for public policy and urban logistics planning, where multiple performance indicators must be simultaneously considered according to different stakeholder perspectives.

The results presented in Table 6 clearly illustrate the trade-offs among the three objectives according to four extreme scenarios, namely minimizing only economic costs, minimizing only environmental costs, minimizing only social costs, and achieving a balanced multi-objective solution. When the model was optimized solely for the economic objective, it achieved the lowest economic cost (100,083.06), but this improvement was accompanied by a substantial increase in social cost (30,984.69). In contrast, prioritizing the social objective drastically reduced the social cost (15,107.20), but resulted in the highest economic (162,885.60) and environmental (41.26) costs. Similarly, optimizing

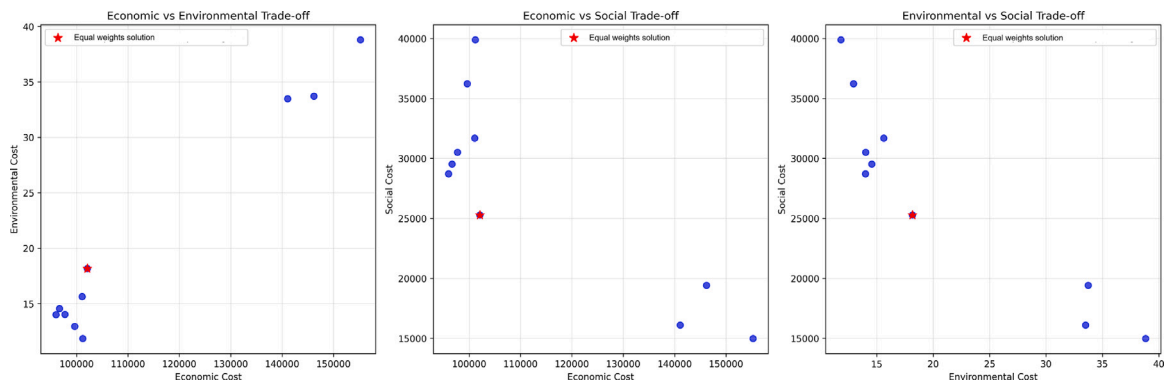


Fig. 5. 2-D Pareto plots.

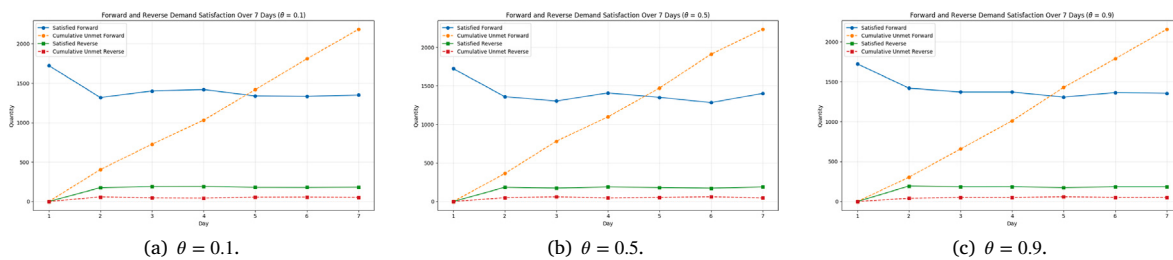


Fig. 6. Comparison of optimal solutions under different values of prioritization weight θ .

exclusively for the environmental objective yielded the best environmental performance (12.68), but at the expense of a considerable increase in social cost (41,702.39).

Meanwhile, the multi-objective scenario with equal weights produced balanced outcomes across all three dimensions (economic: 107,159.85; environmental: 16.98; social: 32,204.29). Although none of the costs reached the optimal values of their respective single-objective cases, this solution represents a compromise that prevents severe deterioration in any individual objective.

The 2D Pareto plots of pairwise objective trade-offs under different weight configurations are presented in Fig. 5. The results show that single-objective settings drive solutions toward extreme corners of the trade-off space, where improvements in the targeted objective come at the expense of the others. By contrast, intermediate configurations generate more balanced outcomes, often lying closer to the solution with equal weights, which serves as a compromise among all three objectives. In general, the results highlight how the optimal routing solutions can adapt to different priorities of decision makers. It should also be noted that each plot represents only two of the three objectives in the optimization problem; therefore, the Pareto fronts are not fully convex. The observed discontinuous patterns result from two-dimensional projections of the underlying three-dimensional trade-off, in which some areas represent joint compromises among all three goals.

From a managerial perspective, the results indicate that strategic prioritization directly influences routing schemes and service performance. Logistics companies, who frequently aim to maximize cost efficiency, may choose the highest weight for economic objective and adopt policies emphasizing revenues accordingly, whereas regulators or other firms prioritizing the sustainable operations may utilize the highest weights for environmental or social objectives. Therefore, the model provides significant support for decision-makers when evaluating sustainability trade-offs under different policy scenarios.

5.3.2. Importance weight of forward relative to reverse demand

To investigate the performance of the algorithm under different prioritization levels between forward and reverse demand, three scenarios

are tested with $\theta = 0.9$ (strong forward priority), $\theta = 0.5$ (balanced), and $\theta = 0.1$ (reverse prioritized). According to the results of satisfied and unmet forward/reverse demand in Fig. 6, the cumulative unmet demand of forward flows remains slightly lower when they are strongly prioritized ($\theta = 0.9$) than in other cases of balanced ($\theta = 0.5$) or reverse ($\theta = 0.1$) prioritization, indicating that assigning higher priority to forward deliveries leads to fewer postponed orders. However, only a limited variation of less than 5% by the end of each period is observed among these scenarios, suggesting that vehicle capacity and time window constraints in the experimental dataset dominate the solution space, thereby diminishing the impact of the weighting parameter. This also highlights that, even though the return rate is significant, the number of returned packages is substantially lower than the forward deliveries. Therefore, returned deliveries can be integrated into the routes with relative ease. From an operational perspective, this finding suggests that demand prioritization should be complemented with dynamic strategies of capacity management rather than being used as a single control mechanism.

5.3.3. Penalty cost for postponement

The sensitivity analysis of penalty costs, ranging from 0 to 200, highlights their influence on accumulated demand satisfaction and unmet demand over the planning horizon. As shown in Fig. 7, higher penalty values consistently reduce unmet forward and reverse demand, with a stronger effect on forward deliveries. When there is no penalty for postponement of orders, unmet forward demand accumulates substantially, since the model has no incentive to prioritize immediate service. As penalties increase, the model allocates more vehicle capacity toward early fulfillment, thereby reducing backlogs. This effect becomes particularly evident beyond a penalty of 70, where both forward and reverse unmet demand decline. Thus, satisfied demand rises, confirming that stricter penalties discourage postponement and prioritize early service completion. These findings emphasize the importance of determining penalty values, as a low penalty could lead to excessive unsatisfied demand, whereas a high penalty enforces strict guarantees of timely service at the expense of operational flexibility.

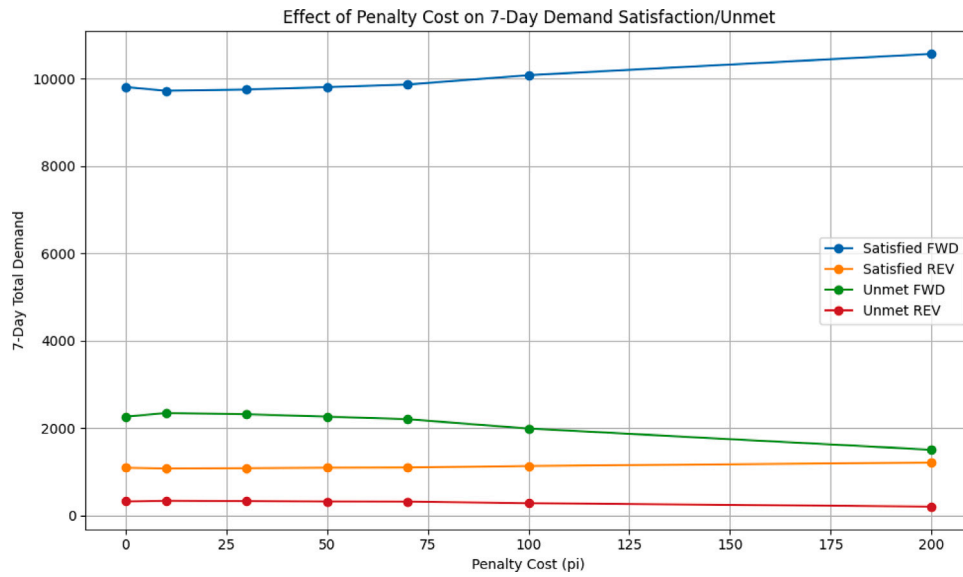


Fig. 7. Total quantity of met/unmet demand against various penalty costs.

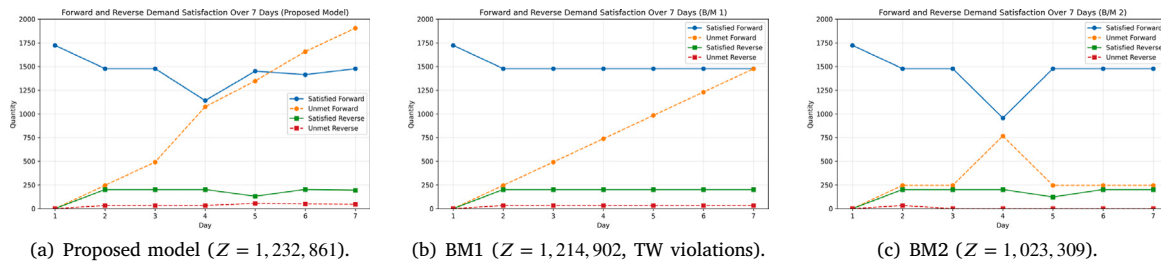


Fig. 8. Comparison of solutions under different carry-over flexibility.

From a managerial perspective, the parameter of penalty cost of delay services can be considered as a specific policy instrument to control a certain level of service reliability. Therefore, penalty tuning enables organizations to align routing strategies with the general service goals as well as customer expectations.

5.3.4. Impact of carry-over flexibility

As allowing unmet demand to be flexibly carried over across operational horizons is a key feature of the proposed rolling-horizon framework, two benchmark scenarios are introduced and compared with the proposed model to assess the operational value of this modeling choice. While the proposed model explicitly allows unmet demand to be postponed and integrated into subsequent routing decisions, benchmark model 1 (B/M 1) is considered to enforce no unmet demand strategy by assigning an extremely high penalty cost, and benchmark model 2 (B/M 2), by contrast, allows unmet demand to occur but prevents it from being served in subsequent planning periods.

Fig. 8 illustrate the resulting demand satisfaction patterns under the three models.

The results indicate that B/M 1 achieves relatively stable daily service performance, as the strong penalty discourages postponement and forces most demand to be served within the same period. However, this rigidity comes at the expense of operational violation of time windows and reduced routing flexibility. Although total unmet demand is lower than in the proposed model, this outcome is achieved by sacrificing feasibility rather than improving resource utilization. From a managerial perspective, B/M 1 represents a strict service policy that may be appropriate for highly time-sensitive goods but can be costly due to much higher penalties, unreliable, and difficult to sustain under demand surges.

In contrast, B/M 2 exhibits a significantly different behavior. Since unmet demand is not carried over, no backlog accumulation is observed across periods. However, this leads to erratic service performance, particularly during peak demand periods, as temporarily unsatisfied requests cannot be compensated in later periods. This lack of coordination and flexibility results in volatile demand satisfaction levels and unreliable services.

Compared with these benchmarks, the proposed model demonstrates a superior balance between short-term operational efficiency and long-term service reliability. By allowing postponement and subsequent recovery of unmet demand, the rolling-horizon approach mitigates extreme service deterioration while preserving routing feasibility. From a managerial standpoint, this flexibility enables operators to address temporary capacity shortages and recover service levels over time, rather than enforcing rigid policies that may incur extremely operational disruptions. Overall, the benchmarking analysis highlights that carry-over mechanisms remain a critical operational strategy for maintaining service continuity and robustness in dynamic e-commerce last-mile logistics.

6. Conclusion

This study proposes a multi-objective mixed-integer programming model and an adaptive large neighborhood search with simulated annealing heuristic for the rolling-horizon pickup and delivery problem with time windows in a two-way logistics system. The approach integrates rolling-horizon planning and interdependent return requests to capture realistic e-commerce operations. Computational experiments on Solomon instances demonstrate that ALNS-SA achieves near-optimal

solutions for small instances with significantly reduced computation times compared to exact methods, and efficiently scales to large instances. The results indicate that coordinated planning of forward and reverse flows under a rolling-horizon framework can effectively maintain load feasibility within capacity constraints, and facilitate proactive strategies for mitigating unmet demand, making the proposed approach well-suited for large-scale, multi-period urban logistics operations with significant return volumes.

The results show that coordinated forward and reverse logistics planning not only improves vehicle utilization but also reduces unmet demand, which is particularly valuable in e-commerce market with high return rates. The sensitivity analysis highlights the critical role of decision parameters such as penalty costs and weight configurations. In particular, the trade-off analysis reveals how prioritizing one sustainability dimension (economic, environmental, or social) inevitably reduces performance in the others, whereas applying balanced weights yield compromise solutions closer to the overall sustainability frontier. These insights enable decision makers to align operational strategies with corporate priorities of whether minimizing costs, reducing emissions, or enhancing service quality, while remaining aware of the trade-offs among such choices.

From a managerial perspective, the findings provide several actionable recommendations for last-mile logistics providers facing increased return flows. First, logistics operators should closely keep track of backlog indicators to trigger adaptive strategies, such as capacity expansion, when service delays begin to propagate across planning periods. Second, penalty policies can be used to effectively balance operational flexibility and service reliability according to customer service goals. Third, prioritization strategies between deliveries and returns should be supported by feasible resource planning to achieve meaningful service improvements.

For future research, two main directions are suggested. First, incorporating uncertainty in demand, travel times, or return flows would better reflect real-world variability and allow for the development of robust or stochastic variants of the model. Second, extending the framework to account for additional operational complexities, such as heterogeneous fleets or multi-depot networks, would further improve its applicability to urban logistics systems. Moreover, future work could also explore strategies to mitigate carry-over effects between horizons, such as dynamic penalty adjustment or anticipatory capacity allocation, to ensure more consistent service performance over time.

CRediT authorship contribution statement

Puong Anh Thi Tran: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Conceptualization.
Patrick Stokkink: Writing – original draft, Validation, Supervision, Methodology, Investigation, Conceptualization.

Data availability

Data will be made available on request.

References

- Alshamrani, A., Mathur, K., & Ballou, R. H. (2007). Reverse logistics: simultaneous design of delivery routes and returns strategies. *Computers & Operations Research*, 34(2), 595–619.
- Angelelli, E., & Mansini, R. (2002). The vehicle routing problem with time windows and simultaneous pick-up and delivery. In *Quantitative approaches to distribution logistics and supply chain management* (pp. 249–267). Springer.
- Arevalo-Ascanio, R., De Meyer, A., Gevaers, R., Guisson, R., & Dewulf, W. (2024). Location-routing problem for integrated supply chain network design with first and last mile: a critical literature review. *Operations and Supply Chain Management: An International Journal*, 17(2), 206–219.
- Bektaş, T., & Laporte, G. (2011). The pollution-routing problem. *Transportation Research Part B: Methodological*, 45(8), 1232–1250.
- Belbağ, S. (2025). A model for vehicle routing problem under returns and emission consideration in B2C E-commerce logistics. *Verimlilik Dergisi, (PRODUCTIVITY FOR LOGISTICS)*, 63–76.
- Braekers, K., Ramaekers, K., & Van Nieuwenhuysse, I. (2016). The vehicle routing problem: State of the art classification and review. *Computers & Industrial Engineering*, 99, 300–313.
- Cardenas, I., Borbon-Galvez, Y., Verlinden, T., Van de Voorde, E., Vanelslander, T., & Dewulf, W. (2017). City logistics, urban goods distribution and last mile delivery and collection. *Competition and Regulation in Network Industries*, 18(1–2), 22–43.
- Çimen, M., & Soysal, M. (2017). Time-dependent green vehicle routing problem with stochastic vehicle speeds: An approximate dynamic programming algorithm. *Transportation Research Part D: Transport and Environment*, 54, 82–98.
- Croes, G. A. (1958). A method for solving traveling-salesman problems. *Operations Research*, 6(6), 791–812.
- de Leeuw, S., Minguela-Rata, B., Sabet, E., Boter, J., & Sigurðardóttir, R. (2016). Trade-offs in managing commercial consumer returns for online apparel retail. *International Journal of Operations & Production Management*, 36(6), 710–731.
- Demir, E., Syntetos, A., & Van Woensel, T. (2022). Last mile logistics: Research trends and needs. *IMA Journal of Management Mathematics*, 33(4), 549–561.
- Deng, S., Li, Y., Zhou, T., & Cao, Y. (2014). Study on recyclable reserve logistics network optimization based on e-commerce. In *2014 international conference on management of e-commerce and e-governance* (pp. 337–340). IEEE.
- Drexli, M. (2012). Synchronization in vehicle routing—a survey of VRPs with multiple synchronization constraints. *Transportation Science*, 46(3), 297–316.
- Dündar, H., Ömürganülşen, M., & Soysal, M. (2021). A review on sustainable urban vehicle routing. *Journal of Cleaner Production*, 285, Article 125444.
- Erdoğan, S., & Miller-Hooks, E. (2012). A green vehicle routing problem. *Transportation Research Part E: Logistics and Transportation Review*, 48(1), 100–114.
- Ge, X., Xue, G., & Wen, P. (2018). Proactive two-level dynamic distribution routing optimization based on historical data. *Mathematical Problems in Engineering*, 2018(1), Article 5191637.
- Govindan, K., Soleimani, H., & Kannan, D. (2015). Reverse logistics and closed-loop supply chain: A comprehensive review to explore the future. *European Journal of Operational Research*, 240(3), 603–626.
- Grewal, D., Iyer, G. R., & Levy, M. (2004). Internet retailing: enablers, limiters and market consequences. *Journal of Business Research*, 57(7), 703–713.
- Griffis, S. E., Rao, S., Goldsby, T. J., & Niranjani, T. T. (2012). The customer consequences of returns in online retailing: An empirical analysis. *Journal of Operations Management*, 30(4), 282–294.
- Hassini, E., Surti, C., & Searcy, C. (2012). A literature review and a case study of sustainable supply chains with a focus on metrics. *International Journal of Production Economics*, 140(1), 69–82.
- Heshmati, S., Verstichel, J., Esprit, E., & Vanden Berghe, G. (2019). Alternative e-commerce delivery policies: A case study concerning the effects on carbon emissions. *EURO Journal on Transportation and Logistics*, 8(3), 217–248.
- Iassinovskaia, G., Limbourg, S., & Riane, F. (2017). The inventory-routing problem of returnable transport items with time windows and simultaneous pickup and delivery in closed-loop supply chains. *International Journal of Production Economics*, 183, 570–582.
- Kahalimoghadam, M., Thompson, R. G., & Rajabifard, A. (2024). Self-adaptive metaheuristic-based emissions reduction in a collaborative vehicle routing problem. *Sustainable Cities and Society*, 110, Article 105577.
- Kokkinaki, A., Dekker, R., de Koster, R., & Pappis, C. (2001). *From e-trash to e-treasure: How value can be created by the new e-business models for reverse logistics. Technical report.*
- Kumar, V., Kumar, V., Brady, M., Garza-Reyes, J. A., & Simpson, M. (2017). Resolving forward-reverse logistics multi-period model using evolutionary algorithms. *International Journal of Production Economics*, 183, 458–469.
- Kuo, R., Wibowo, B., & Zulvia, F. (2016). Application of a fuzzy ant colony system to solve the dynamic vehicle routing problem with uncertain service time. *Applied Mathematical Modelling*, 40(23–24), 9990–10001.
- Lawler, E. L. (1985). *Wiley-interscience series in discrete mathematics, The traveling salesman problem: a guided tour of combinatorial optimization.*
- Li, Y., Guo, H., Wang, L., & Fu, J. (2013). A hybrid genetic-simulated annealing algorithm for the location-inventory-routing problem considering returns under E-supply chain environment. *The Scientific World Journal*, 2013(1), Article 125893.
- Li, J., Zhao, Z., & Cheng, T. (2021). Research on LRP integration of E-commerce logistics under the background of integration of collection and distribution. *Discrete Dynamics in Nature and Society*, 2021(1), Article 6038947.
- Liu, W. (2020). Route optimization for last-mile distribution of rural E-commerce logistics based on ant colony optimization. *IEEE Access*, 8, 12179–12187.
- Ma, H., & Yang, T. (2025). Improved adaptive large neighborhood search combined with simulated annealing (IALNS-SA) algorithm for vehicle routing problem with simultaneous delivery and pickup and time windows. *Electronics*, 14(12), 2375.
- Mallari, C. B., San Juan, J. L., & Bongo, M. (2023). Modelling the vehicle routing problem with delivery and pickup in E-commerce forward-reverse logistics networks based on the triple bottom line framework. *Chemical Engineering Transactions*, 103, 463–468.
- MANIATIS, P. (2023). Exploring the viability of last-mile delivery solutions for sustainable supply chains.

- Min, H. (1989). The multiple vehicle routing problem with simultaneous delivery and pick-up points. *Transportation Research Part A: General*, 23(5), 377–386.
- Moons, S., Braekers, K., Ramaekers, K., Caris, A., & Arda, Y. (2019). The value of integrating order picking and vehicle routing decisions in a B2C e-commerce environment. *International Journal of Production Research*, 57(20), 6405–6423.
- Nepomuceno, N., Saboia, R., & Coelho, A. (2023). A MILP-based very large-scale neighborhood search for the heterogeneous vehicle routing problem with simultaneous pickup and delivery. In *Proceedings of the genetic and evolutionary computation conference* (pp. 330–338).
- Polat, O., Kalayci, C. B., Kulak, O., & Günther, H.-O. (2015). A perturbation based variable neighborhood search heuristic for solving the vehicle routing problem with simultaneous pickup and delivery with time limit. *European Journal of Operational Research*, 242(2), 369–382.
- Qiu, Y., Ni, M., Wang, L., Li, Q., Fang, X., & Pardalos, P. M. (2018). Production routing problems with reverse logistics and remanufacturing. *Transportation Research Part E: Logistics and Transportation Review*, 111, 87–100.
- Ramanathan, R. (2011). An empirical analysis on the influence of risk on relationships between handling of product returns and customer loyalty in E-commerce. *International Journal of Production Economics*, 130(2), 255–261.
- Ramirez, A. M. (2012). Product return and logistics knowledge: Influence on performance of the firm. *Transportation Research Part E: Logistics and Transportation Review*, 48(6), 1137–1151.
- Ropke, S., & Pisinger, D. (2006). An adaptive large neighborhood search heuristic for the pickup and delivery problem with time windows. *Transportation Science*, 40(4), 455–472.
- Shaw, P. (1997). A new local search algorithm providing high quality solutions to vehicle routing problems. vol. 46, Glasgow, Scotland, UK: APES Group, Dept of Computer Science, University of Strathclyde.
- Shaw, P. (1998). Using constraint programming and local search methods to solve vehicle routing problems. In *International conference on principles and practice of constraint programming* (pp. 417–431). Springer.
- Solomon, M. M. (1987). Algorithms for the vehicle routing and scheduling problems with time window constraints. *Operations Research*, 35(2), 254–265.
- Soysal, M., Belbağ, S., & Sel, Ç. (2021). A closed vendor managed inventory system under a mixed fleet of electric and conventional vehicles. *Computers & Industrial Engineering*, 156, Article 107210.
- Soysal, M., Çimen, M., & Belbağ, S. (2020). Pickup and delivery with electric vehicles under stochastic battery depletion. *Computers & Industrial Engineering*, 146, 106512.
- Statista (2020). *Share of online purchases of apparel and shoes returned by consumers in Europe in 2020*. Statista, URL: <https://www.statista.com/statistics/1613605/online-returns-e-commerce-europe/>.
- Tadaros, M., & Migdalas, A. (2022). Bi-and multi-objective location routing problems: Classification and literature review. *Operational Research*, 22(5), 4641–4683.
- Tao, Y., Lin, C., & Wei, L. (2022). Metaheuristics for a large-scale vehicle routing problem of same-day delivery in E-commerce logistics system. *Journal of Advanced Transportation*, 2022(1), Article 8253175.
- Tiwari, K. V., & Sharma, S. K. (2023). An optimization model for vehicle routing problem in last-mile delivery. *Expert Systems with Applications*, 222, Article 119789.
- Toth, P., & Vigo, D. (2014). *Vehicle routing: Problems, methods, and applications*. SIAM.
- Wang, M., Zhang, C., Bell, M. G., & Miao, L. (2022). A branch-and-price algorithm for location-routing problems with pick-up stations in the last-mile distribution system. *European Journal of Operational Research*, 303(3), 1258–1276.
- Wang, J., Zhou, Y., Wang, Y., Zhang, J., Chen, C. P., & Zheng, Z. (2015). Multiobjective vehicle routing problems with simultaneous delivery and pickup and time windows: Formulation, instances, and algorithms. *IEEE Transactions on Cybernetics*, 46(3), 582–594.
- Xiao, J., Li, Y., Cao, Z., & Xiao, J. (2024). Cooperative trucks and drones for rural last-mile delivery with steep roads. *Computers & Industrial Engineering*, 187, Article 109849.
- Yu, X., Li, Y., Liu, J., & Zhou, H. (2024). Solving low-carbon last mile delivery problem using discrete marine predators algorithm. *Applied Soft Computing*, 165, Article 112112.
- Zhang, M. (2017). *Optimal transportation service trading mechanism in e-commerce logistics* (HKU Theses Online (HKUTO)).
- Zhang, M., Pratap, S., Zhao, Z., Prajapati, D., & Huang, G. Q. (2021). Forward and reverse logistics vehicle routing problems with time horizons in B2C e-commerce logistics. *International Journal of Production Research*, 59(20), 6291–6310.
- Zhang, M., Wang, N., He, Z., Yang, Z., & Guan, Y. (2018). Bi-objective vehicle routing for hazardous materials transportation with actual load dependent risks and considering the risk of each vehicle. *IEEE Transactions on Engineering Management*, 66(3), 429–442.
- Zhao, Z., Fang, J., Huang, G. Q., & Zhang, M. (2017). Location management of cloud forklifts in finished product warehouse. *International Journal of Intelligent Systems*, 32(4), 342–370.