

Article

Time-Dependent Vehicle Routing Problem with Simultaneous Pickup-and-Delivery and Time Windows Considering Carbon Emission Costs Using an Improved Ant Colony Optimization Algorithm

Meiling He ¹, Jin Zhang ¹, Xun Han ^{2,*}, Mei Yang ³, Xi Yang ¹, Xiaohui Wu ¹ and Xiaolai Ma ⁴

¹ School of Automotive and Traffic Engineering, Jiangsu University, Zhenjiang 212013, China; hemeiling@ujs.edu.cn (M.H.)

² Intelligent Policing Key Laboratory of Sichuan Province, Sichuan Police College, Luzhou 646000, China

³ College of Automotive Engineering, Guangxi Vocational College of Safety Engineering, Nanning 530100, China

⁴ School of Transportation and Logistics, Southwest Jiaotong University, Chengdu 611756, China

* Correspondence: hanxun@scpolicec.edu.cn

Abstract

In the context of sustainable logistics planning, carbon emission costs have become a critical factor influencing distribution decisions. Meanwhile, the time-dependent characteristics of urban road networks and simultaneous pickup–delivery operations present significant challenges to vehicle routing problems (VRPs). This study addresses a time-dependent vehicle routing problem with simultaneous pickup–delivery and time windows (TDVRPSPDTW). Fuel consumption and carbon emission costs are quantified using a comprehensive emission model, while time-dependent network conditions, simultaneous pickup–delivery demands, and time window constraints are integrated into a unified modeling framework. To solve this NP-hard problem, an improved ant colony optimization (IACO) algorithm is developed by incorporating adaptive large neighborhood search to enhance solution diversity and convergence efficiency. Computational experiments are conducted using internationally recognized VRPSPDTW benchmark datasets and newly constructed TDVRPSPDTW instances, together with sensitivity analyses under varying traffic conditions, time window flexibility, and delivery strategies. The results indicate that the proposed IACO effectively addresses the TDVRPSPDTW. Comparing ant colony optimization with local search (ACO-LS), the IACO achieves a maximum reduction of 11.78% in total distribution cost. Furthermore, relative to the conventional separate pickup–delivery strategy, the simultaneous pickup–delivery mode reduces total distribution cost and carbon emission cost by 49.96% and 53.48%, respectively.

Keywords: vehicle routing problem; time-dependent road network; time windows; simultaneous pickup–delivery; carbon emission; improved ant colony optimization algorithm

Academic Editor: Giulio Erberto Cantarella

Received: 26 December 2025

Revised: 25 January 2026

Accepted: 28 January 2026

Published: 31 January 2026

Copyright: © 2026 by the authors. Submitted for possible open access publication under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

In recent years, the global logistics industry has undergone rapid expansion, largely driven by the rise of e-commerce and the continuous growth in consumer demand [1].

However, the industry still faces several pressing challenges, including persistently high operational costs, inefficient utilization of transportation resources, and low delivery efficiency. According to the latest report released by the International Energy Agency (IEA), China's per capita carbon emissions are currently approximately 16% higher than those of developed economies and nearly twice the global average level [2]. As the second-largest contributor to greenhouse gas emissions, the transportation sector has attracted increasing attention. Within this sector, road transportation accounts for more than 80% of total carbon emissions [3,4]. Reducing carbon emissions in logistics distribution has become a critical issue in the pursuit of sustainable logistics systems. Existing studies indicate that explicitly incorporating carbon emission considerations into logistics distribution planning, together with effective route design and vehicle scheduling, can simultaneously reduce distribution costs and carbon emissions, thereby mitigating the environmental impacts associated with transportation activities [5–7].

From the perspective of operations research and management science, logistics distribution route optimization is commonly formulated as a vehicle routing problem (VRP) [6]. The VRP seeks to determine efficient vehicle routes that satisfy various operational constraints, such as vehicle capacity and customer demand, while achieving objectives such as minimizing the number of vehicles deployed, optimizing delivery time, and reducing overall distribution costs. Dantzig and Ramser [8] first introduced VRPs to optimize the routing of truck fleets transporting fuel between bulk terminals and service stations, aiming to minimize overall transportation costs. Subsequently, Clarke and Wright [9] incorporated practical constraints into the model, establishing the VRP as a fundamental framework for logistics and transportation optimization. Since the 1990s, with the increasing complexity of logistics systems, research on VRPs has expanded considerably and yielded numerous important results. Nevertheless, dynamic demands and world complexities have led to the emergence of various VRP extensions.

Considering customers' time window constraints, Solomon [10] introduced the vehicle routing problem with time windows (VRPTW) and developed a widely used benchmark dataset. Subsequent studies can be classified into hard and soft time window variants. Hard time windows require vehicles to start service strictly within the customer's specified time window, while soft time windows allow for service outside this range by paying a penalty. To capture the time-dependent characteristics of road networks induced by variations in traffic flow, Ichoua et al. [11] introduced the time-dependent vehicle routing problem with time windows (TDVRPTW), in which the delivery horizon is divided into discrete time intervals, and vehicle speeds vary accordingly to reflect dynamic traffic conditions. Fan et al. [12] further utilized historical traffic data and approximated the time-dependent travel times using trigonometric functions to generate smooth representations of the dynamic road network.

To promote sustainable development, carbon emissions have attracted increasing attention in both academia and practice. To more accurately estimate fuel consumption and emission levels, Demir et al. [13] proposed a comprehensive emission model. This model was subsequently adopted by Majidi et al. [14] and Qin et al. [15] to incorporate carbon costs into routing optimization. Chen et al. [16] further extended this approach to cold chain logistics and proposed a time-dependent green vehicle routing problem (TDGVRP) from a low-carbon economic perspective. Considering heterogeneous customer demands, such as the one-hour pickup-and-delivery requirements in e-commerce instant delivery services, the vehicle routing problem with simultaneous pickup–delivery and time windows (VRPSPDTW) has gained increasing research interest. Wang and Chen [17] formulated a VRPSPDTW model with an objective function defined as a weighted sum of travel distance and fleet size, and developed a cooperative co-evolutionary genetic algorithm.

Wang et al. [18] further developed a multi-commodity mixed-integer programming model incorporating reverse logistics and packaging recovery.

Although the above studies have achieved substantial progress, most existing VRP formulations consider only a subset of practical features encountered in real-world logistics operations, such as time-dependent traffic conditions, carbon emissions, or simultaneous pickup–delivery requirements. To address these limitations, this study integrates the core features of the TDVRPTW, TDGVRP, and VRPSPDTW into a unified framework and introduces a novel theoretical framework for the time-dependent vehicle routing problem with simultaneous pickup–delivery and time windows (TDVRPSPDTW). Specifically, it incorporates time-dependent functions to characterize dynamic variations in road networks, adopts a simultaneous pickup–delivery mode to address heterogeneous customer demands, and employs soft time windows as service constraints. Moreover, the model integrates carbon emission costs, computed using a comprehensive emission model, to overcome the limitations of previous VRP studies that neglected realistic factors such as time-dependent traffic conditions and simultaneous pickup–delivery requirements. Based on these considerations, a model that integrates these practical features is formulated and solved using an IACO algorithm enhanced with adaptive large neighborhood search to avoid premature convergence. The main contributions of this research are as follows:

- This study proposes a novel TDVRPSPDTW model, which simultaneously accounts for carbon emissions, time-varying vehicle speeds, and customer requests for both pickup-and-delivery services. The model is developed by integrating fixed, transportation, carbon emission, and time window penalty costs, addressing the gaps in prior research.
- An IACO algorithm is designed to solve the proposed TDVRPSPDTW. The algorithm initializes pheromone trails using a savings-based heuristic, modifies the state transition rule by introducing distance-saving values, time window deviation factors, and time window width factors, and incorporates a large neighborhood search strategy to destruct and repair local optima. Furthermore, an operator scoring mechanism is introduced for adaptive selection of removal and insertion operators.
- Extensive computational experiments are conducted on standard international VRP-SPDTW benchmark datasets of varying scales. Comparative analyses with representative algorithms demonstrate the superior performance of the proposed approach. In addition, sensitivity analyses on constructed TDVRPSPDTW instances confirm that the use of soft time windows and simultaneous pickup–delivery strategies can effectively enhance efficiency and reduce overall costs.

The remainder of this paper is organized as follows. Section 2 reviews the related literature. Section 3 formulates the mathematical model of the TDVRPSPDTW. Section 4 presents the improved ant colony optimization algorithm. Section 5 provides computational results and sensitivity analysis. Finally, Section 6 concludes the paper and outlines the limitations together with potential avenues for future research.

2. Literature Review

This section reviews the related literature from three perspectives: the objectives, model formulations, and solution algorithms of the vehicle routing problem. A comparative summary is provided at the end.

2.1. Objectives

2.1.1. Classical VRPs

In classical vehicle routing problem studies, the primary objective was to design cost-efficient routes for a truck fleet transporting gasoline from bulk terminals to multiple service stations [8]. Subsequent studies have explored various optimization objectives. Some focus on minimizing total delivery time or vehicle arrival time [19–21], while others aim to minimize the number of vehicles or total route length [17,22–24]. Over time, minimizing the overall distribution cost through the integration of multiple weighted objectives has become a primary optimization strategy. Srivastava et al. [25] and Cai et al. [26] formulated multi-objective VRPTW models that consider vehicle utilization, total travel distance, maximum route duration, early arrival waiting time, and late service penalties. Qi et al. [27] incorporated total vehicle usage time, energy consumption, and customer satisfaction into a multi-objective model. Wang et al. [28] analyzed the impact of transportation resource sharing on reducing fleet size and maintenance costs, thereby minimizing overall logistics expenditure.

2.1.2. VRPs with Carbon Emission

With the increasing emphasis on sustainable development, a growing body of research has incorporated environmental considerations into vehicle routing optimization, in which the accurate modeling of fuel consumption and carbon emissions has become an important topic in green logistics. Most existing studies are based on the fundamental assumption that carbon emissions are approximately proportional to fuel consumption; accordingly, fuel consumption is explicitly modeled to indirectly quantify the carbon emission costs generated by transportation activities and incorporated into the objective function of the vehicle routing problem.

From the perspective of emission modeling approaches, the related literature can generally be classified into two categories: macroscopic empirical models and microscopic comprehensive models. On the one hand, some studies estimate fuel consumption using linear fuel consumption rate formulations, where fuel usage is calculated per unit travel distance, or adopt the MEET model proposed by the European Commission to assess the energy consumption and emissions of heavy-duty vehicles from a traffic engineering perspective [29]. Owing to their modeling simplicity and computational efficiency, these approaches have been widely embedded into multi-objective and time-dependent green vehicle routing problems to balance transportation efficiency and environmental impacts. On the other hand, with increasing demands for modeling accuracy, a growing body of research has employed comprehensive emission models that capture the effects of operational characteristics—such as engine speed, vehicle load, engine displacement, and friction coefficients—on fuel consumption at a microscopic level [13,30]. Based on these models, carbon emission costs are subsequently calculated, enabling more refined and realistic emission estimations compared with macroscopic approaches. Moreover, in specific application contexts, particularly in cold-chain logistics, comprehensive emission models have been further extended to account for both fuel consumption during vehicle operations and indirect carbon emissions arising from refrigeration equipment [16]. By integrating these additional emission sources into routing optimization, existing studies have broadened the scope of low-carbon vehicle routing research.

In summary, although existing studies have extensively investigated both classical VRP objectives and emerging environmental objectives, certain limitations remain. In particular, the joint consideration of time-dependent traffic conditions, fuel-based carbon emission costs, and service-related penalty costs within a unified optimization framework has not been sufficiently explored. To address these gaps, this study integrates fixed

vehicle costs, transportation costs, carbon emission costs derived from fuel consumption, and time window penalty costs into a single objective function aimed at minimizing the total distribution cost. Carbon emissions under time-dependent road networks are quantified based on fuel consumption using a comprehensive emission model, thereby contributing to a more holistic and realistic optimization of vehicle routing problems.

2.2. Model Formulations

2.2.1. VRPTW

Time window constraints represent one of the earliest and most influential extensions of the classical vehicle routing problem. By introducing customer-specific service intervals, Solomon [10] formulated the vehicle routing problem with time windows (VRPTW), substantially enhancing its practical relevance. Subsequent studies have explored alternative formulations of time window constraints. Some have imposed strict service time constraints, requiring that deliveries be completed within the designated time windows [21,31–34], whereas others introduced soft time windows that permit early or late service with corresponding penalty costs to improve model flexibility [25,26,35].

2.2.2. TDVRPTW

Building upon the VRPTW, the time-dependent vehicle routing problem with time windows (TDVRPTW) was proposed to explicitly account for the temporal variability of urban traffic conditions by incorporating time-related road network constraints into vehicle routing models [36]. To capture traffic dynamics, Ichoua et al. [11] proposed a piecewise constant-speed model, dividing the planning horizon into discrete intervals, each with a constant vehicle speed, to approximate traffic variations. In addition, some studies have utilized continuous speed functions derived from historical urban traffic data through functional approximation to describe smooth temporal variations in vehicle travel speeds [12,37].

2.2.3. VRPSPDTW

With the rapid development of e-commerce and reverse logistics, customers increasingly exhibit simultaneous pickup-and-delivery demands. The vehicle routing problem with simultaneous pickup–delivery and time windows (VRPSPDTW) allows vehicles to perform both pickup-and-delivery operations at the same customer location, which not only accommodates heterogeneous customer requirements but also improves vehicle load utilization and reduces empty travel [38,39]. For instance, Liu et al. [40] introduced reverse logistics into the VRPSPDTW framework, while Wang et al. [18] further incorporated product returns and packaging recovery through a multi-commodity mixed-integer programming model to enhance resource utilization efficiency. Nevertheless, these studies did not incorporate time-dependent road network characteristics into their formulations.

In summary, prior research has successively introduced time windows, time-dependent traffic conditions, and simultaneous pickup–delivery requirements into vehicle routing models, leading to a variety of VRP extensions such as VRPTW, TDVRPTW, and VRPSPDTW. However, studies that systematically integrate these constraints within a unified time-dependent framework remain limited. Building upon existing research, this study jointly incorporates time-dependent road networks, soft time windows, and heterogeneous pickup–delivery demands, thereby further extending the modeling scope and practical applicability of vehicle routing problem research.

2.3. Solution Algorithms

Early research on vehicle routing problems primarily relied on exact algorithms, which are capable of guaranteeing global optimality. However, their computational complexity grows exponentially with problem size, rendering them impractical for large-scale applications [41,42].

In contrast, metaheuristic algorithms such as ant colony optimization (ACO), genetic algorithms (GAs), and adaptive large neighborhood search (ALNS) draw inspiration from natural evolution and collective intelligence. These methods provide a good balance between exploration and exploitation and can efficiently produce high-quality near-optimal solutions for large-scale and complex problems [24]. However, single algorithms often face performance bottlenecks under complex constraints, motivating the development of hybrid metaheuristics that combine the strengths of multiple paradigms.

Against such background, increasing research attention has been devoted to the vehicle routing problem with simultaneous pickup-and-delivery and time windows. Wang and Chen [17] formulated a VRPSPDTW model with the objective defined as a weighted sum of travel distance and the number of vehicles, and proposed a co-evolutionary genetic algorithm for its solution. More importantly, by extending classical Solomon benchmark instances, they constructed a set of VRPSPDTW test instances with different scales and characteristics, which has since served as a standardized and representative experimental platform for subsequent algorithmic studies. Based on these benchmark instances, a variety of advanced metaheuristic approaches have been developed and validated, including an adaptive large neighborhood search algorithm with route recombination strategy [43], a two-stage algorithm [44], and an ant colony optimization algorithm with a destruction–repair strategy [45], thereby further advancing solution methodologies for the VRPSPDTW.

Among various metaheuristic approaches, ant colony optimization (ACO) has been widely applied to vehicle routing problems and their extensions due to its cooperative search mechanism and favorable extensibility. To address the tendency of ACO to suffer from premature convergence under complex constraints, existing studies have incorporated various local improvement strategies into the ACO framework to regulate the balance between exploration and exploitation. Current enhancement efforts mainly focus on the construction of heuristic initial solutions, the refinement of pheromone updating strategies, and the integration of ACO with genetic algorithms, adaptive large neighborhood search, or simulated annealing to improve computational efficiency and solution stability [46,47]. In addition, Wu et al. [45] systematically investigated the effects of key parameters, such as the number of ants and the pheromone evaporation rate, on algorithmic performance, providing useful guidance for parameter tuning in ACO-based routing applications. Overall, ACO and its hybrid variants have become commonly adopted solution approaches for VRPs and their extensions, with recent research emphasizing structural enhancements and adaptive mechanisms to improve applicability and solution quality in combinatorial optimization problems [30].

Building upon the aforementioned research advances, this study develops an improved ant colony optimization (IACO) algorithm for the TDVRPSPDTW. Within the classical ACO framework, the pheromone initialization and updating mechanisms are refined, and an adaptive large neighborhood destruction–repair strategy is incorporated to enhance search diversity and alleviate premature convergence under complex constraints, thereby further improving overall solution performance.

Table 1 summarizes representative studies on vehicle routing problems from the perspectives of network condition, optimization objectives, distribution modes, and resolution. The comparison highlights the distinctions and innovations of this study relative to the existing literature. Here, NV is the number of vehicles used, TD the total travel

distance, CEC the carbon emission cost, and TPC the time window penalty cost. In particular, CEC indicates whether carbon emission cost is explicitly calculated based on fuel consumption, rather than aggregated proxies or additional emission sources such as refrigeration systems. As shown in Table 1, most existing studies focus on only a subset of practical features encountered in real-world logistics operations, such as neglecting time-dependent road networks, omitting explicit carbon emission costs, or insufficiently modeling heterogeneous customer demands. As a result, the ability of these models to comprehensively capture the complexity of realistic logistics systems remains limited.

Table 1. Classification of related papers.

Reference	Network Condition	Objectives				Distribution Modes	Resolution
		NV	TD	CEC	TPC		
Yu et al. [48]	Static	-	√	√	-	Delivery	ALNS
Liu et al. [49]	Time-dependent	-	√	-	-	Pickup and delivery	Ant colony system and virtual transformation method
Chen et al. [16]	Time-dependent	√	√	√	√	Delivery	Hybrid simulated annealing algorithm
Erdogdu and Karabulut [50]	Static	-	√	√	-	Delivery	Hybrid-ALNS
Wen et al. [51]	Static	√	√	√	-	Delivery	Adaptive large neighborhood search
Wang et al. [28]	Static	√	√	-	√	Pickup and delivery	Genetic and particle swarm optimization algorithm
Ren et al. [47]	Static	√	√	√	√	Pickup and delivery	Improved ant colony optimization
Luo et al. [52]	Time-dependent	-	-	√	-	Delivery	Branch-price-and-cut algorithm
Ahmed et al. [53]	Static	√	√	-	-	Delivery	Modified football game algorithm
Wu et al. [45]	Static	√	√	-	-	Pickup and delivery	An ant colony optimization algorithm with destroy and repair strategies
Praxedes et al. [54]	Static	√	√	-	-	Pickup and delivery	Branch-cut-and-price algorithm
Zhao et al. [23]	Time-dependent	√	√	-	-	Delivery	Time-dependent split algorithm
Ren et al. [6]	Static	√	√	√	-	Delivery	Improved genetic algorithm
Chen et al. [46]	Static	-	√	-	√	Delivery	Improved genetic ant colony optimization algorithm
Nyako et al. [29]	Time-dependent	-	√	√	√	Delivery	Non-dominated sorting genetic Algorithm enhanced with machine learning
This paper	Time-dependent	√	√	√	√	Pickup and delivery	Improved ant colony optimization

Note: √ indicates this factor is considered in the paper; - indicates it is not.

3. TDVRPSPDTW Description and Formulation

3.1. Problem Description

The time-dependent vehicle routing problem with simultaneous pickup–delivery and time windows (TDVRPSPDTW) investigated is defined as follows. In the distribution network $G = (C_0, E)$, $C_0 = \{0, 1, 2, 3, \dots, c\}$ denotes the set of all nodes, $E = \{(i, j) | i, j \in C_0, i \neq j\}$ represents the set of edges connecting nodes (i, j) , and $C = \{1, 2, 3, \dots, c\}$ denotes the set of customer nodes excluding the depot (denoted as Node 0). Vehicles depart from the depot with an initial load of goods. Upon reaching a customer

node, they first complete the delivery operation and then perform the pickup task before traveling to the next customer node. This process continues until all customer nodes in the distribution network have been served with simultaneous pickup-and-delivery operations. A schematic representation of the TDVRPSPDTW distribution routes is provided in Figure 1.

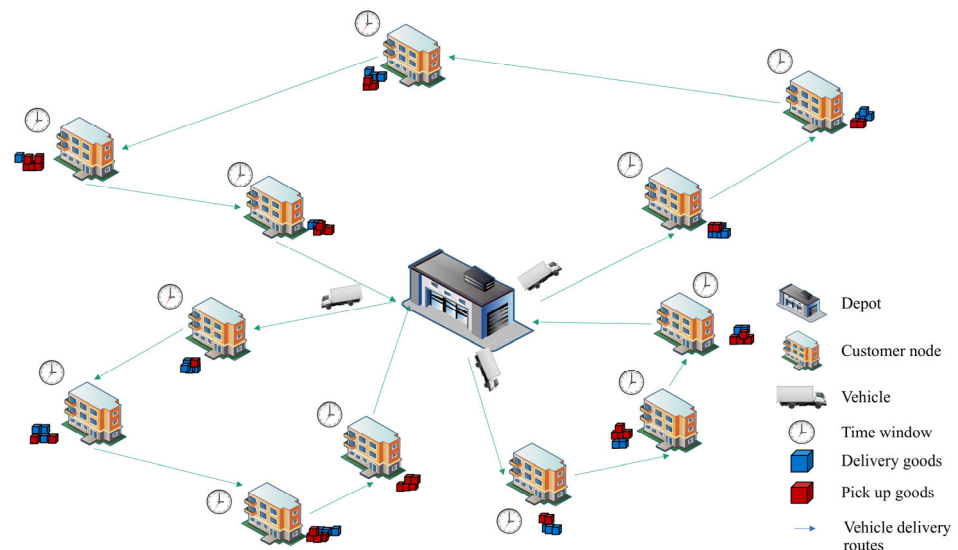


Figure 1. Schematic diagram of TDVRPSPDTW distribution routes.

In this study, a soft time window constraint is adopted. Under this mechanism, vehicles are permitted to serve customers even if they arrive before or after the specified time window, but a penalty cost is incurred accordingly. Assuming that the service all customer demands must be satisfied, the routing plan is formulated with the objective of minimizing the total distribution cost.

The problem investigated is formulated under the following assumptions:

- (1) The distribution network comprises a single depot. Vehicles depart from the depot to perform their assigned services and are required to return to the depot with a load equal to the aggregate pickup demand of the customer nodes served. The return must occur prior to the designated depot opening time.
- (2) Each vehicle can provide simultaneous pickup-and-delivery services to multiple customer nodes. However, each customer node can be visited by only one vehicle exactly once, implying that customer demand is indivisible and cannot be split among multiple vehicles.
- (3) For customer nodes with simultaneous pickup-and-delivery demands, the service operation follows a “delivery-first, pickup-later” sequence, whereby delivery is completed prior to loading the items to be picked up. The volume of goods is not considered in this study.
- (4) The maximum load capacity of all vehicles is assumed to be identical and known a priori. Overloading is strictly prohibited during the distribution process.
- (5) For each customer, the location, delivery demand, pickup demand, and time window information are known. The delivery-and-pickup demands of a single customer are both smaller than the vehicle’s maximum capacity.
- (6) Vehicles are allowed to arrive outside the specified time window; however, such violations incur the corresponding penalty costs.
- (7) The distribution road network exhibits time-dependent characteristics, reflected by variations in vehicle travel speed across different time intervals.

- (8) During service at a customer node, the vehicle engine is deactivated, thereby generating no fuel consumption or carbon emissions.

The parameter definitions are summarized in Table 2.

Table 2. Parameter definitions in mathematical models.

Parameter	Definitions
C	Customer nodes, $C = \{1, 2, 3, \dots, c\}$
C_0	All nodes, $C_0 = C \cup \{0\}$, 0 represents the depot
E	Set of arcs, where $E = \{(i, j) i, j \in C_0, i \neq j\}$
K	Set of all vehicles, $K = \{1, 2, 3 \dots k\}$
T	Set of all time intervals, $T = \{1, 2, 3, \dots, m\}$
f_1	Fixed unit departure cost of a vehicle (CNY)
f_2	Unit transportation cost per unit time of a vehicle (CNY/h)
f_3	Unit carbon emission cost of a vehicle (CNY/kg)
f_4	Unit time penalty cost for a vehicle's early arrival at a customer node (CNY/h)
f_5	Unit time penalty cost for a vehicle's late arrival at a customer node (CNY/h)
s_j	Delivery demand of customer node j , $j \in C$ (kg)
q_j	Pickup demand of customer node j , $j \in C$ (kg)
d_{ij}	Distance from node i to node j , $i, j \in C_0$ (km)
U	Maximum load capacity of a vehicle (kg)
m_0	Unloaded weight (dead weight) of a vehicle (kg)
h_i	Service time at customer node i , $i \in C$ (h)
$[e_i, l_i]$	Preset time window of customer node i , $i \in C$
$[e_0, l_0]$	Opening hours of the depot
A_m	Start time of the m th time interval
B_m	End time of the m th time interval
v_{ijk}^m	Travel speed of vehicle k from node i to node j within the m th time interval, $i, j \in C_0$ (km/h)
d_{ijk}^m	Travel distance of vehicle k from node i to node j within the m th time interval, $i, j \in C_0$ (km)
t_{ijk}^m	Travel time of vehicle k from node i to node j within the m th time interval, $i, j \in C_0$ (h)
t_{ijk}	Total travel time of vehicle k from node i to node j , $i, j \in C_0$ (h)
e	Carbon emission factor (CO ₂ /kg)
E_{ijk}^m	Carbon emissions of vehicle k from node i to node j within the m th time interval, $i, j \in C_0$ (kg)
y_{ijk}^m	Decision variable: equals 1 if vehicle k travels from node i to node j within the m th time interval, and 0 otherwise
Lt_{ik}	Time when vehicle k departs from customer node i
At_{ik}	Time when vehicle k arrives at customer node i
W_{ijk}	Load of vehicle k when traveling from node i to node j (kg)
ψ	Calorific value of fuel (L/kJ)
ζ	Engine friction coefficient (kJ/rev/L)
Z	Engine speed (rev/s)
V	Engine displacement (L)

ε	Transmission coefficient
δ	Air resistance coefficient

3.2. Mathematical Formulation

The time-dependent function proposed by Ichoua et al. [11] is adopted to characterize the dynamic properties of the road network. The instantaneous vehicle speed within a very short time interval is approximated as a constant value, and the continuous travel time of a vehicle is indirectly obtained by aggregating the segmented travel speeds, as illustrated in Figure 2.

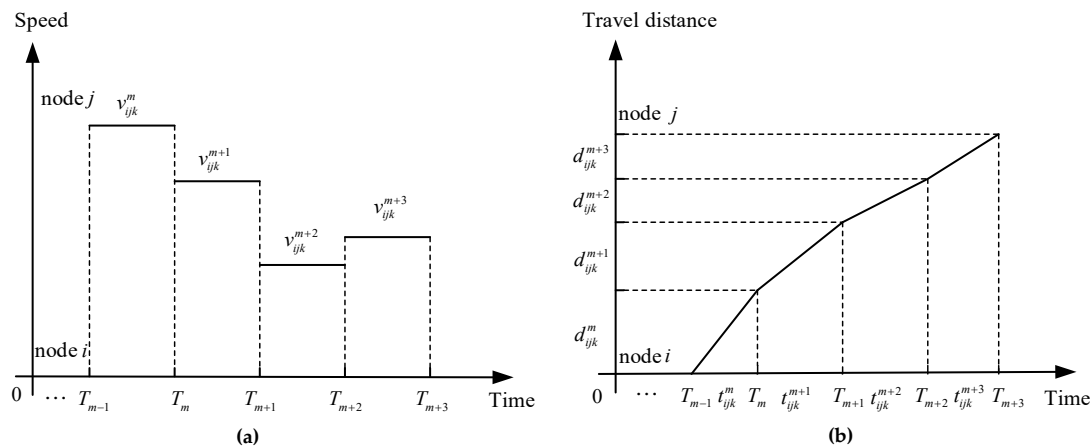


Figure 2. Discrete speeds and continuous traveling times of vehicles: (a) Speed at different times; (b) Driving distance during different time periods.

Specifically, the operating time horizon of the depot is divided into M equal intervals, each of duration D . Let $T = \{T_0, T_1, \dots, T_M\}$ denote the set of discrete time points, and $[T_{m-1}, T_m]$ represent the m th interval. When vehicle k travels from customer node i to customer node j , let $Lt_{ik} \in [T_{m-1}, T_m]$ denote its departure time, v_{ijk}^m the travel speed during the interval, d_{ijk}^m the travel distance, and t_{ijk}^m the travel time. Based on the relationship between the known distance d_{ij} (from node i to node j) and d_{ijk}^m , there are two scenarios: non-cross-interval travel and cross-interval travel. For cross-interval travel ($d_{ijk}^m < d_{ij}$), the remaining distance for vehicle k to travel from node i to node j after the m th time interval is $R_{ij}^m = d_{ij} - d_{ijk}^m$. In summary, the total travel time t_{ijk} of vehicle k from node i to node j is calculated in two steps:

Step 1: Calculate the travel time within the initial interval. $d_{ijk}^m = v_{ijk}^m (T_m - Lt_{ik})$, if $d_{ijk}^m > d_{ij}$, set $t_{ijk}^m = d_{ij} / v_{ijk}^m$, $t_{ijk} = t_{ijk}^m$, terminate the calculation and output t_{ijk} ; otherwise $R_{ij}^m = d_{ij} - d_{ijk}^m$, $t_{ijk}^m = T_m - Lt_{ik}$, $m = m + 1$.

Step 2: Calculate the travel time within subsequent intervals. $d_{ijk}^m = v_{ijk}^m D$, if $d_{ijk}^m < R_{ij}^{m-1}$, set $R_{ij}^m = R_{ij}^{m-1} - d_{ijk}^m$, $t_{ijk}^m = D$, $m = m + 1$, repeat step 2; otherwise $t_{ijk}^m = R_{ij}^m / v_{ijk}^m$, $t_{ijk} = \sum_{m \in T} t_{ijk}^m$, and terminate the calculation and output t_{ijk} .

Under the above piecewise time-dependent speed formulation, the first-in-first-out (FIFO) property of the road network is preserved. Specifically, a vehicle departing later from the origin node of an arc cannot arrive earlier at its destination node than a vehicle departing earlier. As demonstrated by Ichoua et al. [11], this FIFO property holds for any

arc under the adopted time-dependent travel time calculation, thereby ensuring temporal consistency and feasibility of the proposed TDVRPSPDTW model.

A comprehensive emission model is adopted to estimate vehicle fuel consumption from a microscopic perspective. It incorporates the impacts of parameters such as vehicle load, travel speed, air resistance, engine speed, displacement, and friction coefficient to approximate the fuel consumption per second of the vehicle. The estimated fuel consumption is then converted into the corresponding carbon emission values, which are subsequently multiplied by the unit cost of carbon emissions to obtain the overall carbon emission cost. Meanwhile, considering the time-dependent characteristics of the road network, the travel speed of vehicle k on route segment (i, j) varies across different time intervals. Accordingly, the calculation of fuel consumption adopts a cumulative approach based on time intervals. Specifically, the fuel consumption O_{ijk}^m of vehicle k traveling from node i to node j during the m th time interval is computed using Equation (1).

$$O_{ijk}^m = \psi \zeta Z V d_{ijk}^m / v_{ijk}^m + \psi \varepsilon g(m_0 + W_{ijk}) d_{ijk}^m + \psi \varepsilon \delta d_{ijk}^m (v_{ijk}^m)^2 \quad (1)$$

where ψ is the calorific value of fuel, ζ is the engine friction coefficient, Z is the engine speed (rev/s), V is the engine displacement, ε is the transmission coefficient, m_0 is the unloaded weight of a vehicle, W_{ijk} is the load of vehicle k when traveling from node i to node j , and δ is the air resistance coefficient.

Subsequently, the corresponding carbon emissions are obtained from O_{ijk}^m by applying the carbon emission factor e , as expressed in Equation (2).

$$E_{ijk}^m = e O_{ijk}^m \quad (2)$$

Based on the characteristics of the problem addressed, the total distribution cost is defined as the sum of fixed vehicle costs, transportation costs, carbon emission costs, and time window penalty costs. Accordingly, a mixed-integer programming model is formulated with the objective of minimizing the total distribution cost, as presented below:

$$\begin{aligned} \min TC = & f_1 \sum_{k \in K} \sum_{j \in C} \sum_{m \in T} y_{0jk}^m + f_2 \sum_{k \in K} \sum_{i, j \in C_0} \sum_{m \in T} t_{ijk}^m y_{ijk}^m + f_3 \sum_{k \in K} \sum_{i, j \in C_0} \sum_{m \in T} E_{ijk}^m y_{ijk}^m \\ & + f_4 \sum_{k \in K} \sum_{i \in C_0} \sum_{j \in C} \sum_{m \in T} y_{ijk}^m \max \{ (e_j - Lt_{ik} - t_{ijk}^m), 0 \} \\ & + f_5 \sum_{k \in K} \sum_{i \in C_0} \sum_{j \in C} \sum_{m \in T} y_{ijk}^m \max \{ (Lt_{ik} + t_{ijk}^m - l_j), 0 \} \end{aligned} \quad (3)$$

s. t.

$$\sum_{k \in K} \sum_{i \in C_0} \sum_{m \in T} y_{ijk}^m = 1, \forall j \in C \quad (4)$$

$$\sum_{i \in C_0} \sum_{m \in T} y_{ijk}^m - \sum_{i \in C_0} \sum_{m \in T} y_{jik}^m = 0, \forall j \in C, \forall k \in K \quad (5)$$

$$\sum_{j \in C} \sum_{m \in T} y_{0jk}^m \leq 1, \forall k \in K \quad (6)$$

$$\sum_{j \in C} W_{0jk} = \sum_{i, j \in C_0} \sum_{m \in T} y_{ijk}^m s_j, \forall k \in K \quad (7)$$

$$\sum_{i \in C} W_{i0k} = \sum_{i, j \in C_0} \sum_{m \in T} y_{jik}^m q_i, \forall k \in K \quad (8)$$

$$0 \leq W_{ijk} \leq U, \forall i, j \in C_0, \forall k \in K \quad (9)$$

$$\sum_{i \in C_0} \sum_{m \in T} y_{ijk}^m (W_{ijk} - s_j) = \sum_{i \in C_0} \sum_{m \in T} y_{jik}^m (W_{jik} - q_j), \forall j \in C, \forall k \in K \quad (10)$$

$$A_m y_{ijk}^m \leq L t_{ik} y_{ijk}^m \leq B_m y_{ijk}^m, \forall i, j \in C_0, \forall k \in K, \forall m \in T \quad (11)$$

$$t_{ijk} = \sum_{m \in T} t_{ijk}^m y_{ijk}^m, \forall i, j \in C_0, \forall k \in K \quad (12)$$

$$L t_{ik} = A t_{ik} + h_i, \forall i \in C_0, \forall k \in K \quad (13)$$

$$\sum_{m \in T} y_{ijk}^m (A t_{ik} + h_i + t_{ijk}^m) \leq A t_{jk}, \forall i, j \in C_0, \forall k \in K \quad (14)$$

$$L t_{ik} + t_{i0k} \leq l_0, \forall i \in C, \forall k \in K \quad (15)$$

$$y_{ijk}^m \in \{0, 1\}, \forall i, j \in C_0, \forall k \in K, \forall m \in T \quad (16)$$

Equation (3) defines the objective function, which minimizes the total distribution cost comprising fixed vehicle costs, transportation costs, carbon emission costs, and time window penalty costs. Equation (4) enforces that each customer node is visited exactly once, with indivisible demand, and ensures that delivery-and-pickup requests are simultaneously satisfied. Equation (5) stipulates that once a vehicle has arrived at and completed service for a customer node, it must subsequently depart from that node, while Equation (6) guarantees that each vehicle is utilized at most once. Equations (7) and (8) serve as demand fulfillment constraints. Specifically, Equation (7) requires that the vehicle load upon departure from the depot equals the total delivery demand of the customer nodes to be served, whereas Equation (8) requires that the vehicle load upon returning to the depot equals the aggregate pickup demand of the customer nodes already served. Equation (9) restricts the vehicle load on any arc from exceeding its maximum capacity, and Equation (10) maintains load balance before and after servicing a node. Equation (11) ensures that $L t_{ik}$, the departure time of vehicle k from customer node i , lie within the corresponding time interval $[A_m, B_m]$. Equation (12) introduces the calculation of travel time across different time intervals and is used to determine the travel time of vehicle k travels from node i to node j . Equation (13) specifies that the departure time $L t_{ik}$ of vehicle k from node i is equal to the sum of its arrival time $A t_{ik}$ and the service time h_i . Equation (14) enforces the temporal precedence constraint, ensuring that the arrival time $A t_{jk}$ at node j is not earlier than the arrival time at the preceding node i , plus the service time h_i at node i and the travel time between nodes i and j . Equation (15) ensures that vehicles return to the depot before the latest allowable depot closing time. Finally, Equation (16) defines the attributes of the decision variables.

4. Solution Methodology

This section presents three significant enhancements to the classical ACO algorithm aimed at capturing the unique features of the TDVRPSPDTW. First, the savings heuristic is employed to construct high-quality initial solutions, which are further utilized to initialize the pheromone trails. Second, the state transition rules are refined by incorporating problem-specific indicators, namely the distance-saving value, the time window deviation factor, and the time window width factor, to better guide the search process. Third, the

ALNS strategy is integrated as a local search mechanism to enhance the exploration capability of the algorithm and reduce the risk of premature convergence. The details of these three enhancements are presented in Sections 4.1–4.4.

4.1. Initial Pheromone Initialization

One notable limitation of the classical ACO algorithm is the absence of predefined initial pheromone values, which may lead to slow convergence. To address this issue, the present study employs the savings heuristic to generate initial solutions, which are subsequently used for pheromone initialization, as expressed in Equation (17). In this equation, τ_0 represents the initial pheromone value, n denotes the total number of customer nodes to be served, and D_0 represents the total travel distance of the initial solution generated through the savings heuristic.

$$\tau_0 = \frac{1}{nD_0} \quad (17)$$

The procedure for constructing the initial solution using the savings heuristic is described as follows:

1. Initialization: Construct independent routes for all customer nodes, each traveling directly to and from the depot, i.e., assign one vehicle per customer node.
2. Savings calculation: Compute the distance savings resulting from merging every pair of routes, and rank these values in descending order.
3. Route merging: Iteratively evaluate the ranked savings list to identify feasible route pairs that satisfy the vehicle load capacity constraint. Merge the first feasible pair, update the savings list, and repeat the process. If no feasible pair remains, the procedure terminates.

4.2. Improved State Transition Rules

The state transition rule is the core mechanism of the ACO algorithm, determining how ants sequentially select the next target node in constructing routes. Following prior research [55], this study adopts a pseudo-random proportional rule that integrates both probabilistic exploration and deterministic exploitation. Specifically, the algorithm compares a random number $q \in [0,1]$ with a predefined control parameter q_0 to decide whether the next node is selected deterministically or probabilistically.

To further enhance the quality of distribution plans, three additional factors are incorporated into the state transition rule: the distance-saving value g_{ij} , the time window deviation impact factor div_j , and the time window width impact factor spa_j . These enhancements encourage the selection of customer nodes associated with shorter travel distances, smaller penalties for early or late arrivals, and narrower time windows.

The distance-saving value is calculated using Equation (18). The principle is as follows: consider assigning customer nodes i and j to two separate routes, then compute the difference between this combined route distance and the direct distance between nodes i and j . d_{ij} denotes the direct distance between customer node i and customer node j . A larger g_{ij} indicates a greater likelihood that customer node j will be chosen as the next destination by the ant.

$$g_{ij} = d_{i0} + d_{0j} - d_{ij} \quad (18)$$

Because the TDVRPSPDTW problem allows soft time window violations, div_j is introduced to reduce penalty costs. The degree of deviation is given in Equation (19). If At_j

, the arrival time at customer node j falls within $[e_j, l_j]$, there is no time window deviation, and div_j is set to 1.

$$div_j = \begin{cases} e_j - At_j, & At_j < e_j \\ 1, & e_j \leq At_j \leq l_j \\ At_j - l_j, & At_j > l_j \end{cases} \quad (19)$$

Customer nodes differ in the width of their service windows, defined as the difference between the latest and earliest allowable service times. A narrower window implies less scheduling flexibility, making such customers more critical to prioritize. This factor helps guide ants toward serving tighter time windows first, thereby improving vehicle utilization and routing efficiency.

Based on these considerations, the improved state transition rule for an ant moving from node to node is expressed in Equations (20) and (21).

$$j^k = \begin{cases} \arg \max_{j \in PE_i^k} \left[(\tau_{ij})^\sigma \left(\frac{g_{ij}}{d_{ij}} \right)^\varsigma \left(\frac{1}{div_j} \right)^\pi \left(\frac{1}{spa_j} \right)^\omega \right], & q \leq q_0 \\ p_{ij}^k, & q > q_0 \end{cases} \quad (20)$$

$$p_{ij}^k = \begin{cases} \frac{(\tau_{ij})^\sigma \left(\frac{g_{ij}}{d_{ij}} \right)^\varsigma \left(\frac{1}{div_j} \right)^\pi \left(\frac{1}{spa_j} \right)^\omega}{\sum_{h \in PE_i^k} (\tau_{ih})^\sigma \left(\frac{g_{ih}}{d_{ih}} \right)^\varsigma \left(\frac{1}{div_h} \right)^\pi \left(\frac{1}{spa_h} \right)^\omega}, & j \in PE_i^k \\ 0, & j \notin PE_i^k \end{cases} \quad (21)$$

where τ_{ij} denotes the pheromone concentration between node i and node j ; σ , ς , π , and ω represent the coefficients of pheromone importance factor, visibility importance factor, time window deviation factor, and time window width impact factor, respectively; p_{ij}^k denotes the state transition probability for ant k selecting the next node j from node i ; and PE_i^k represents the set of unserved customer nodes that satisfy both the vehicle load capacity constraint and the time window constraint when ant k is located at node i .

4.3. ALNS Local Search

Compared with traditional local search strategies such as 2-opt, insertion, and swap, the ALNS framework incorporates multiple destruction and repair operators. During the iterative process, the algorithm adaptively updates operator weights based on their historical performance, thereby selecting more effective combinations of destruction and repair operators to explore the solution space. This mechanism not only increases the likelihood of finding high-quality solutions but also expands the search neighborhood.

The ALNS strategy is used to intensify the search around the initial solutions generated by the ACO algorithm. It effectively mitigates the tendency of ACO to become trapped in local optima, thereby improving overall solution quality. The specific destruction and repair operators utilized are detailed as follows:

4.3.1. Destruction Operators

(1) Similarity destruction operator

The implementation procedure is as follows: randomly select a customer node i from the current distribution plan, remove it, and place it in the deletion list RL . According to Equation (22), compute the similarity $X(i, j)$ between customer node i and each undeleted customer node by considering four factors: travel distance, time window, pickup–delivery demand, and whether they are on the same route. Sort the customer nodes in ascending order of similarity, then delete the $\text{rand}^a |RL| - th$ customer node (where rand is a random number within $[0, 1)$, a is a position parameter balancing randomness and similarity selection, and $|RL|$ is the size of the deletion list RL). Randomly select a deleted customer node from the deletion list RL , and repeat the above operation until RL is empty.

$$X(i, j) = \lambda_1 \frac{d_{ij}}{\max_{i, j \in C} (d_{ij})} + \lambda_2 \frac{|s_i - s_j| + |q_i - q_j|}{\max_{i, j \in C} (|s_i - s_j| + |q_i - q_j|)} + \lambda_3 \frac{|e_i - e_j| + |l_i - l_j|}{\max_{i, j \in C} (|e_i - e_j| + |l_i - l_j|)} + \lambda_4 G \quad (22)$$

where λ_1 , λ_2 , λ_3 , and λ_4 are the weight coefficients for travel distance, time window difference, pickup–delivery demand difference, and whether nodes are on the same route between customer nodes i and j , respectively, with $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$; G equals 1 if nodes i and j are on the same route, and 0 otherwise.

(2) Worst-cost destruction operator

In each iteration, calculate the reduction in the objective function caused by deleting each unremoved customer node from the current solution. Sort all unremoved customer nodes in descending order according to their corresponding objective cost reduction values. Select the customer node with the largest objective cost reduction for deletion. Repeat this process iteratively until the number of deleted customer nodes reaches RL .

(3) Worst-route destruction operator

During each iteration, compare the number of customers in each route, select the route with the fewest customers for deletion, and then use a repair operator to reinsert the removed nodes into more suitable positions.

(4) Random destruction operator

In each iteration, randomly select customer nodes from the current distribution plan and delete them, storing the removed nodes sequentially in the deletion list RL , until a destroyed solution with $|RL|$ removed customer nodes is obtained.

4.3.2. Repair Operators

(1) Greedy repair operator

Under the premise of satisfying all constraints, the optimal insertion position and the corresponding insertion cost for each customer node in the deletion list RL are calculated within the current distribution plan. Customer nodes are then sorted in ascending order of insertion cost. The node with the smallest insertion cost is inserted at its optimal position, and subsequently removed from RL . If no feasible insertion exists under the constraints, the number of vehicles is incremented by one, and the node is inserted into a new route. After updating the deletion list RL , the insertion costs for the remaining nodes are recalculated, and the sorting and insertion procedure is repeated until all nodes in RL are reinserted.

(2) Regret value repair operator

Under the premise of satisfying all constraints, the regret value for each customer node in RL is calculated. The regret value is defined as the difference between the insertion cost at the optimal position and that at the second-best position. Customer nodes are then sorted in descending order of regret value, and the node with the largest regret value is inserted at its optimal position. After insertion, the node is removed from RL , and the deletion list is updated. This process is repeated until RL is empty.

4.3.3. Operator Selection Strategy

In the ALNS framework, the selection probability θ_i of each operator is computed based on the operator weights for the six destruction and repair operators described above. A roulette-wheel selection mechanism is employed to choose a pair of destruction and repair operators in each iteration, with the selection probability determined by θ_i . The formula for θ_i is given in Equation (23), where μ_i denotes the weight of operator i , and r is the total number of operators.

$$\theta_i = \frac{\mu_i}{\sum_{j=1}^r \mu_j}, \quad i = 1, 2, \dots, r \quad (23)$$

At the beginning of the iterative process, all operators are assigned equal initial weights. One destruction operator is selected using the roulette-wheel method to disrupt the current solution, followed by the selection of a repair operator to restore the solution, producing a repaired solution. Operator performance is recorded using the following scoring rules.

- If the repaired solution is better than the current solution and constitutes a global optimum, the operator is awarded α_1 points.
- If the repaired solution is better than the current solution but not a global optimum, the operator is awarded α_2 points.
- If the repaired solution is accepted as the current solution according to a Metropolis acceptance criterion, the operator is awarded α_3 points.

During the search, the weight μ_i of operator i is dynamically updated at the end of each period K according to Equation (24), based on the cumulative score and number of times the operator has been applied. Specifically, μ_i is updated only after each period K rather than continuously. In Equation (24), φ denotes the operator weight update coefficient, π_i denotes the cumulative score of operator i within a given update period, and n_i denotes the cumulative number of calls of operator i within the same given update period.

$$\mu_i = (1 - \varphi)\mu_i + \varphi\pi_i/n_i \quad (24)$$

4.4. Solution Process

Figure 3 illustrates the overall solution process of the proposed IACO algorithm. The advantages of the proposed algorithm can be summarized as follows:

First, the savings-based initialization strategy generates high-quality initial solutions, which guide the search toward promising regions of the solution space.

Second, the modified state transition rule jointly considers distance-saving and time window deviation, thereby enhancing solution guidance under time-dependent constraints.

Finally, the embedded adaptive large neighborhood search mechanism improves diversification and repair capability, allowing the algorithm to escape local optima and achieve stable convergence.

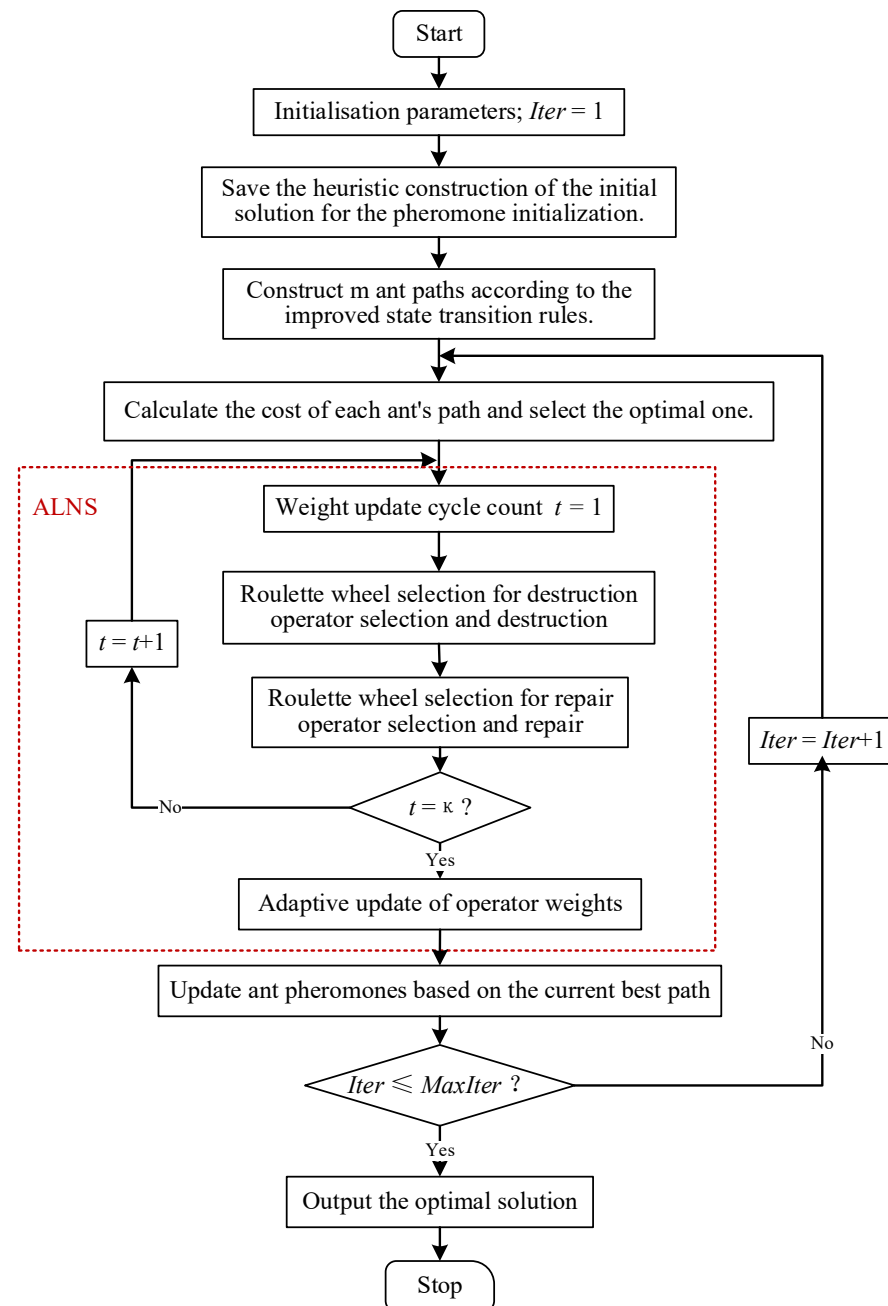


Figure 3. Flowchart of improved ant colony optimization algorithm.

5. Experimental Analysis

5.1. Parameter Settings

By reviewing the relevant literature and materials [15,45,56,57], the parameters related to the model are set as follows: the fixed unit departure cost of a vehicle $f_1 = 400$ CNY, the unit transportation cost per unit time of a vehicle $f_2 = 180$ CNY/h, the

unit carbon emission cost of a vehicle $f_3 = 0.05$ CNY/kg, the carbon emission factor $e = 3.1$ CO₂/kg, the unloaded weight of the vehicle $m_0 = 6350$ kg, the calorific value of fuel $\psi = 3.06 \times 10^{-5}$ L/kJ, the engine friction coefficient $\zeta = 0.2$ kJ/rev/L, the engine speed $Z = 33$ rev/s, the engine displacement $V = 5$ L, the transmission coefficient $\varepsilon = 2.77 \times 10^{-3}$, the air resistance coefficient $\delta = 1.65$, the gravitational constant $g = 9.81$ m/s², and the unit time penalty costs for violating the time window are $f_4 = 60$ CNY/h (early arrival) and $f_5 = 120$ CNY/h (late arrival), respectively.

The algorithm code was written in MATLAB R2019b and was run in an environment equipped with an AMD Ryzen 7 6800H processor with Radeon graphics (32 GB of RAM) and the Windows 11 operating system. By reviewing relevant literature and materials and adjusting parameters for optimization [38,45], the algorithm parameters are set as follows: the number of ants $m = 50$, the maximum number of iterations $MaxIter = 100$, the transfer control parameter $q_0 = 0.5$, the coefficient of pheromone importance factor $\sigma = 2$, the coefficient of visibility importance factor $\eta = 1$, the coefficient of time window deviation factor $\pi = 3$, the coefficient of time window width impact factor $\omega = 2$, the constant pheromone intensity $Q = 1000$, the pheromone evaporation factor $\rho = 0.2$, the size of the deletion list (i.e., the number of customer nodes to be deleted) $|RL| = [0.15n, 0.25n]$, the weight coefficients in similarity destruction operator $\lambda_1 = 0.4, \lambda_2 = 0.3, \lambda_3 = 0.2, \lambda_4 = 0.1$, the scoring rules $\alpha_1 = 15, \alpha_2 = 8, \alpha_3 = 3$, the operator weight update period $\kappa = 50$, and the operator weight update coefficient $\varphi = 0.25$.

5.2. Verification of Algorithm Effectiveness

The TDVRPSPDTW investigated is an extension of the classical VRPSPDTW. To evaluate the effectiveness of the proposed IACO algorithm, we employ internationally recognized benchmark instances of VRPSPDTW. These instances were generated by Wang and Chen [17], who introduced pickup demands into the well-known VRPTW benchmark set originally proposed by Solomon. The dataset consists of 65 instances, including 9 small-scale cases with 10, 25, or 50 customers, and 56 large-scale cases with 100 customers. The 100-customer set is further divided into 6 categories: C1, C2, R1, R2, RC1, and RC2. Specifically, in R-type instances, customer locations are randomly distributed; in C-type instances, customer locations exhibit a clustered distribution; and RC-type instances combine both random and clustered spatial patterns, representing a mixed distribution structure. Given the heterogeneity in instance size and structure, the objective function is defined with a lexicographic priority: first, to minimize the number of vehicles, and second, to minimize the total travel distance. The proposed IACO is tested on all 9 small-scale instances (10, 25, and 50 customers) and on 18 representative large-scale instances with 100 customers.

Table 3 reports a comparative analysis of IACO against CPLEX [22], discrete cuckoo search (DCS) [58], and memetic algorithm with efficient local search and extended neighborhood (MATE) [59] on small-scale instances. Here, NV denotes the number of vehicles used, and TD represents the total travel distance. Due to its limited computational capability, CPLEX is only able to solve five instances: RCdp1001, RCdp1004, RCdp1007, RCdp2501, and RCdp5001. For these cases, IACO obtains solutions identical to those of CPLEX in terms of both NV and TD. Compared to the MATE algorithm, IACO produces slightly inferior results on the RCdp5004 instance, with a difference of less than 0.02%. For all other instances, the results of both methods are identical. In comparison with DCS, the

solutions of IACO are also consistent for all but one case, RCdp5004, where the total travel distance differs by only 1.07%.

Table 3. Comparison of IACO performance on VRPSPDTW instances with 10, 25, and 50 customers.

Instance/Number of Customers	CPLEX		DCS		META		IACO	
	NV	TD	NV	TD	NV	TD	NV	TD
RCdp1001/10	3	348.98	3	348.98	3	348.98	3	348.98
RCdp1004/10	2	216.69	2	216.69	2	216.69	2	216.69
RCdp1007/10	2	310.81	2	310.81	2	310.81	2	310.81
RCdp2501/25	5	551.05	5	551.05	5	551.05	5	551.05
RCdp2504/25	7 *	738.32 *	4	473.46	4	473.46	4	473.46
RCdp2507/25	7 *	634.20 *	5	540.87	5	540.87	5	540.87
RCdp5001/50	9	994.18	9	944.18	9	944.18	9	944.18
RCdp5004/50	14 *	1961.53 *	6	725.59	6	733.21	6	733.32
RCdp5007/50	13 *	1814.33 *	7	809.72	7	809.72	7	809.72

Note: * indicates exceeding the memory value.

Table 4 presents a comparison of the VRPSPDTW instances with 100 customers against three metaheuristics: co-evolution genetic algorithm (CO-GA) [17], DCS [58], and sine cosine and firefly perturbed sparrow search algorithm (SFSSA) [60]. Specifically, compared with CO-GA, the proposed IACO reduces the number of vehicles by one in two instances (Rdp110 and RCdp203). Among the remaining 16 instances with the same number of vehicles, IACO achieves equal or superior travel distances in 11 cases, with the maximum improvement reaching 4.65%. Relative to DCS, IACO also reduces the vehicle count by one in instance Rdp110. For the 17 instances with equal vehicle numbers, IACO attains equal or better travel distances in 12 cases, with a maximum improvement of 4.30%. In comparison with SFSSA, IACO reduces the number of vehicles in three instances (RCp110, RCdp105, and RCdp203). Among the 15 instances where the number of vehicles is identical, IACO delivers equal or superior travel distances in 9 cases, achieving up to 5.40% improvement.

Table 4. Comparison of IACO performance on VRPSPDTW instances with 100 customers.

Instance	CO-GA		DCS		SFSSA		IACO	
	NV	TD	NV	TD	NV	TD	NV	TD
Rdp102	17	1488.04	17	1490.13	17	1485.85	17	1490.08
Rdp107	11	1087.95	11	1084.00	11	1081.90	11	1094.80
Rdp110	12	1116.99	12	1108.81	12	1120.88	11	1135.09
Rdp205	3	1064.43	3	1051.38	3	1016.54	3	1015.71
Rdp206	3	961.32	3	957.81	3	902.11	3	916.63
Rdp211	3	839.61	3	819.88	3	806.04	3	805.40
Cdp105	11	983.10	11	981.45	11	983.10	11	983.10
Cdp106	11	878.29	11	878.29	11	878.29	11	878.29
Cdp109	10	940.49	10	940.49	10	985.08	10	931.90
Cdp202	3	591.56	3	591.56	3	591.56	3	591.56
Cdp204	3	590.60	3	590.60	3	590.60	3	590.60
Cdp207	3	588.29	3	588.29	3	588.29	3	588.29
RCdp101	15	1652.90	15	1654.32	15	1654.84	15	1667.57
RCdp105	14	1581.26	14	1581.26	15	1578.78	14	1573.60
RCdp108	11	1175.04	11	1170.12	11	1169.84	11	1203.64
RCdp203	4	964.65	3	1087.37	4	987.15	3	1064.83
RCdp204	3	822.02	3	822.02	3	845.55	3	800.89

RCdp206	3	1176.85	3	1166.88	3	1124.5	3	1194.97
---------	---	---------	---	---------	---	--------	---	---------

These results clearly demonstrate the effectiveness of the proposed IACO in solving VRPSPDTW, highlighting its ability to achieve competitive or superior performance compared with other advanced metaheuristics.

5.3. TDVRPSPDTW Instance Analysis

Currently, no benchmark instances are available for the TDVRPSPDTW problem. Therefore, based on the characteristics of the problem, we construct test instances by extending the VRPSPDTW instances with 10, 25, and 50 customers introduced in Section 5.1. Specifically, the time window constraints of customer nodes are relaxed, allowing vehicles to serve customers either earlier or later than the specified time windows. In addition, the operating horizon of the depot is evenly divided into five periods: $[0, 0.2l_0)$, $[0.2l_0, 0.4l_0)$, $[0.4l_0, 0.6l_0)$, $[0.6l_0, 0.8l_0)$, and $[0.8l_0, l_0]$. The time-dependent functions corresponding to four typical traffic scenarios proposed by Andres Figliozzi [42] are summarized in Table 5. According to the relative position of the vehicle departure time with respect to morning and evening peak periods, the speed profiles are classified into four types, denoted as conditions a–d.

Condition a: The vehicle departs during the morning peak period, resulting in low travel speeds in the initial stage due to severe congestion. In the second time interval, traffic demand decreases and vehicle speed increases accordingly. The third interval corresponds to the midday period, during which congestion intensifies again. In the fourth interval, occurring before the evening peak, traffic conditions gradually improve. The final interval coincides with the evening peak, leading to a significant reduction in travel speed.

Condition b: The vehicle departs at an early time when congestion has not yet occurred. The second interval corresponds to the morning peak, during which travel speed decreases. Traffic conditions are alleviated during the midday interval, allowing speeds to recover. In the subsequent interval, the vehicle encounters the evening peak and travel speed decreases again. In the final interval, congestion dissipates as the evening peak ends, and travel speed increases.

Condition c: The vehicle also departs early and is not affected by the morning peak during the first three intervals. The morning peak occurs in the last two intervals, where congestion has a pronounced impact on travel speed.

Condition d: The vehicle departs during the morning peak period, resulting in congested traffic conditions and low travel speeds in the first two intervals. As the morning peak subsides in the remaining intervals, traffic conditions improve and vehicle speed increases accordingly.

For urban logistics distribution, urban arterial roads serve as the primary traffic corridors and are designed with operating speeds in the range of 40–60 km/h. Consistent with assumptions commonly adopted in the related literature [12,52], an average speed of 50 km/h is adopted as the baseline speed in the construction-and-computational-experiment instance of this study.

Table 5. Figliozzi's speed time dependence function.

Type	Travel Speed
a	TD1a=[1.00 1.60 1.05 1.60 1.00]
	TD2a=[1.00 2.00 1.50 2.00 1.00]
	TD3a=[1.00 2.50 1.75 2.50 1.00]
b	TD1b=[1.60 1.00 1.05 1.00 1.60]

c	TD2b=[2.00 1.00 1.50 1.00 2.00]
	TD3b=[2.50 1.00 1.75 1.00 2.50]
	TD1c=[1.60 1.60 1.05 1.00 1.00]
	TD2c=[2.00 2.00 1.50 1.00 1.00]
	TD3c=[2.50 2.50 1.75 1.00 1.00]
d	TD1d=[1.00 1.00 1.05 1.60 1.60]
	TD2d=[1.00 1.00 1.50 2.00 2.00]
	TD3d=[1.00 1.00 1.75 2.50 2.50]

5.3.1. Algorithm Performance Comparison

To evaluate the performance of the proposed IACO on the TDVRPSPDTW instances, a comparison is conducted against the ant colony optimization with local search (ACO-LS). The ACO-LS algorithm extends the classical ACO by incorporating insertion, swap, and 2-opt operators for local search, while adopting the same state transition rules as those used in this study. The basic parameter settings of ACO-LS are kept consistent with those of the IACO. Each instance is independently solved 10 times. The comparative results are summarized in Table 6, where Best represents the best total distribution cost among the 10 runs, Avg denotes the average total distribution cost, SD indicates the standard deviation, Gap refers to the deviation in the best total distribution cost between IACO and ACO-LS, and Time denotes the average computational time in seconds.

Table 6. Performance comparison of IACO and ACO-LS for TDVRPSPDTW instances.

Instance	ACO-LS				IACO				Gap (%)
	Best	Avg	SD	Time	Best	Avg	SD	Time	
RCdp1001	1903.95	2035.02	232.40	18.94	1903.99	1903.99	0.00	21.70	0.00%
RCdp1004	1588.68	1588.68	0.00	17.97	1566.71	1566.71	0.00	23.75	−1.38%
RCdp1007	1734.53	1764.87	20.20	18.30	1695.80	1695.80	0.00	23.78	−2.23%
RCdp2501	3504.43	3912.63	310.11	69.91	3504.43	3508.31	4.55	108.28	0.00%
RCdp2504	3445.66	3566.63	214.64	72.02	3450.05	3493.14	44.12	106.25	0.13%
RCdp2507	3630.52	3841.11	213.66	70.60	3596.30	3615.80	66.96	103.01	−0.94%
RCdp5001	6959.93	7343.34	371.14	232.47	6140.32	6570.14	271.45	428.72	−11.78%
RCdp5004	5531.57	5616.87	48.09	235.37	4944.77	5013.71	79.75	400.36	−10.61%
RCdp5007	5973.28	6157.94	288.49	230.25	5475.33	6006.12	256.03	420.05	−8.34%

As shown in Table 6, for TDVRPSPDTW instances of different scales, the proposed IACO consistently achieves lower best and average distribution costs compared to ACO-LS. The average relative deviation of the best solutions is 4.23%, and this advantage becomes more pronounced as the number of customers increases, reaching a maximum relative deviation of 11.78% when the customer size is 50. Moreover, in terms of solution stability, the standard deviations obtained by IACO are generally smaller. In terms of computational time, both IACO and ACO-LS exhibit increasing solution times as the problem scale grows. Although IACO requires additional computational effort due to its enhanced search mechanisms, the solution times of both algorithms remain within a practically acceptable range for all tested instances. Figure 4 illustrates the comparison of carbon emission cost (CEC) and total cost (TC) obtained from the best solutions generated by IACO and ACO-LS across different instances. Compared with ACO-LS, the proposed IACO achieves a maximum reduction of 20.68% in carbon emission cost. Taken together, these results demonstrate the effectiveness of the proposed IACO in achieving high-quality and stable solutions across different instance sizes.

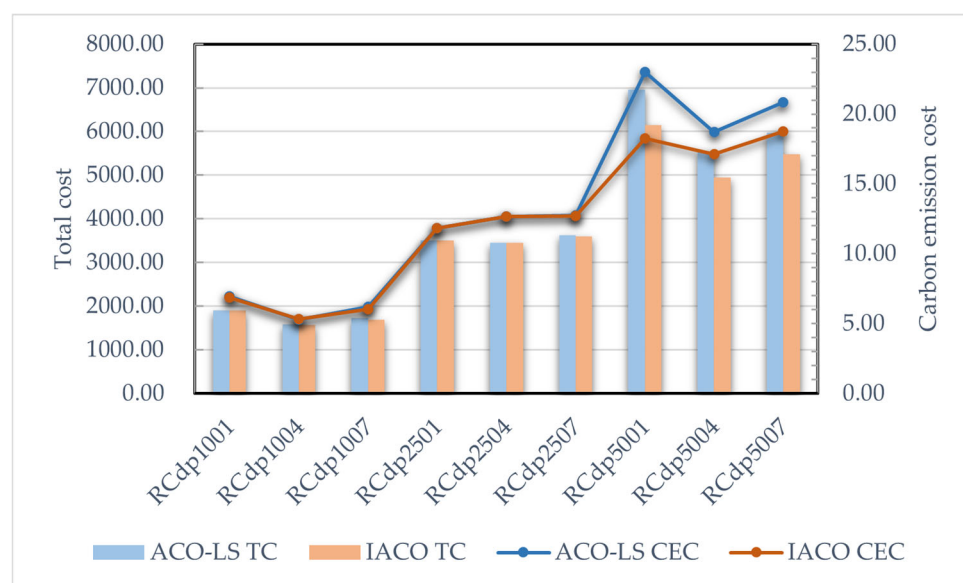


Figure 4. CEC and TC of IACO and ACO-LS under different TDVRPSPDTW instances.

Furthermore, for the instance with 100 customer nodes, instance Rcdp101 is selected as a representative case while keeping all other conditions unchanged. Both the ACO-LS and IACO algorithms are independently executed 10 times. The results indicate that, compared with the best distribution cost of 13,921.15 obtained by ACO-LS, the best cost achieved by the IACO algorithm is 12,125.72, corresponding to a reduction of approximately 12.90%. Figures 5 and 6 present the iterative convergence curves of ACO-LS and IACO, respectively, for solving instance Rcdp101. It can be observed that ACO-LS converges after approximately 90 iterations, whereas IACO begins to converge in fewer than 20 iterations. Compared with ACO-LS, the proposed IACO exhibits faster improvement in solution quality during the iterative search process and more stable convergence behavior.

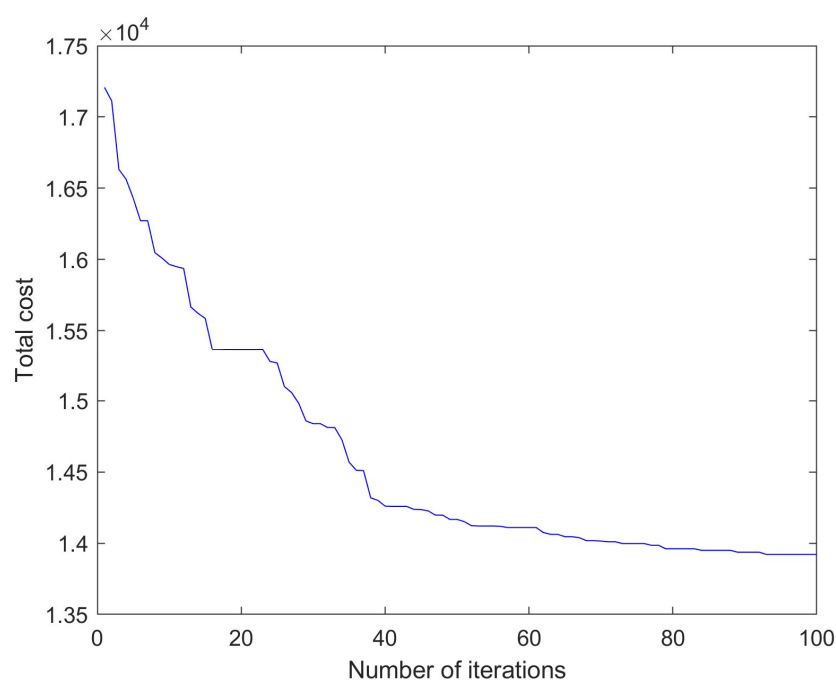


Figure 5. Iterative convergence curve of ACO-LS solving Rcdp101 instance.

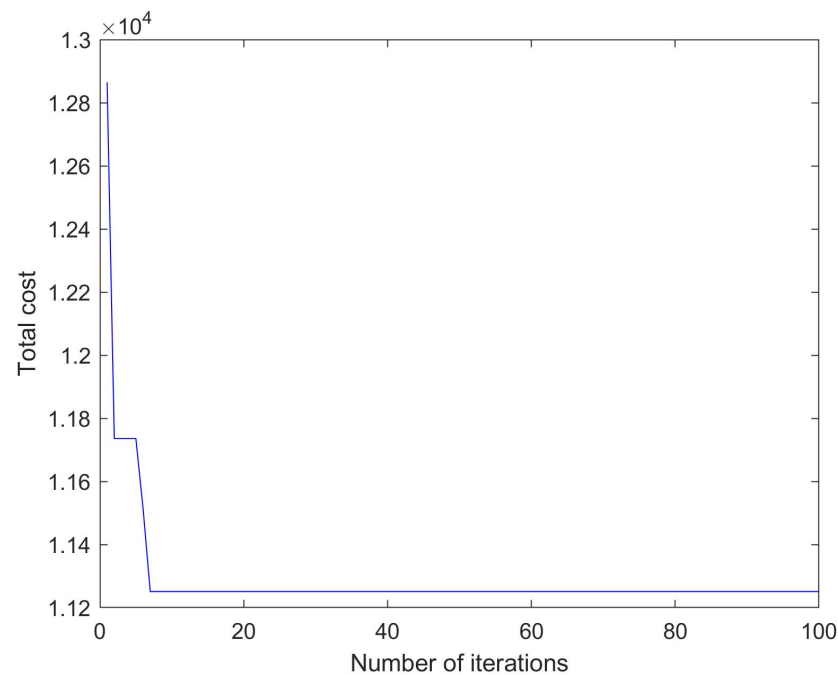


Figure 6. Iterative convergence curve of IACO solving Rcdp101 instance.

Furthermore, under identical constraint conditions and parameter settings, the proposed model is solved with respect to three different optimization objectives. Specifically, for each objective, 10 independent runs are conducted and the best solution is selected for analysis. The three optimization modes are defined as follows: Mode 1: minimization of the total travel distance; Mode 2: minimization of carbon emission cost; Mode 3: minimization of the total distribution cost.

Here, CEC represents carbon emission cost, TC denotes the total distribution cost. The corresponding results are reported in Table 7. It can be observed that, for each instance, the proposed algorithm is able to obtain the optimal solution under all three objective settings. However, when a single objective formulation is adopted, the total distribution cost increases significantly compared with the integrated model proposed in this study. On average, the total cost increases by 6.15% and 5.41%, respectively, while the maximum increases reach 14.87% and 8.67%. This indicates that minimizing travel distance or carbon emission cost alone generally requires sacrificing economic efficiency. In contrast, the proposed model simultaneously accounts for vehicle fixed costs, travel costs, fuel consumption, and carbon emission costs, enabling a balanced trade-off among multiple factors and achieving the minimum total distribution cost. As a result, the model provides an effective decision support tool for logistics enterprises to enhance distribution efficiency and promote sustainable development.

Table 7. Solution results for various optimization objectives.

Instance	Mode 1			Mode 2			Mode 3		
	TD	CEC	TC	TD	CEC	TC	TD	CEC	TC
RCdp5001	703.41	17.22	6398.66	706.54	17.18	6240.64	745.81	18.25	6140.32
RCdp5004	690.86	17.23	5614.51	692.22	16.94	5112.65	699.57	17.11	4944.77
RCdp5007	740.89	18.13	5500.49	756.12	18.03	5499.58	760.38	18.73	5475.33

5.3.2. Influence of Time-Dependent Road Network Characteristics

To investigate the impact of time-dependent road network characteristics on solution quality, computational experiments are conducted on the constructed benchmark instance

RCdp5004. Under identical conditions and constraints, 10 independent runs are performed for each scenario, and the best solution is selected for analysis. Using the IACO algorithm, we compared distribution schemes under seven distinct network conditions (Case 1–Case 7). Specifically, Case 1–Case 4 correspond to four sets of time-dependent functions ($TD1a=[1.00 \ 1.60 \ 1.05 \ 1.60 \ 1.00]$, $TD1b=[1.60 \ 1.00 \ 1.05 \ 1.00 \ 1.60]$, $TD1c=[1.60 \ 1.60 \ 1.05 \ 1.00 \ 1.00]$, $TD1d=[1 \ 1 \ 1.05 \ 1.60 \ 1.60]$) proposed by Andres Figliozzi [42], with velocity parameters identical to those specified earlier. Case 5 represents a static network with an average vehicle speed equal to that of case 1, while Case 6 and Case 7 denote static networks with constant speeds of 40 km/h and 60 km/h, respectively. The results for RCdp5004 under these network conditions are reported in Table 8, where TT denotes the total vehicle travel time.

Table 8. Solution results of RCdp5004 instance under varying road network conditions.

Network Conditions	NV	TD	TT	CEC	TC
Case 1	6	699.57	809.67	17.11	4944.77
Case 2	6	724.98	827.39	17.70	5014.01
Case 3	6	700.09	821.95	17.44	5043.81
Case 4	6	738.24	861.57	17.26	5084.45
Case 5	6	733.91	884.23	17.61	5188.34
Case 6	8	815.13	1216.62	20.62	6994.84
Case 7	6	707.27	707.27	16.56	4787.06

As shown in Table 8, even when the average speed is the same, the distribution schemes differ across networks with distinct time-dependent properties. When comparing Case 2–Case 5 with Case 1, the number of vehicles used remains constant at 6, while travel distance deviates by 3.63%, 0.07%, 5.53%, and 4.91%, respectively; travel time deviates by 2.19%, 1.52%, 6.41%, and 9.21%, respectively; carbon emission cost deviates by 3.45%, 1.93%, 0.88%, and 2.92%, respectively; and delivery cost deviates by 1.40%, 2.00%, 2.82%, and 4.93%. Overall, although the deviations are relatively minor, they confirm that different time-dependent functions yield slightly different solutions. Considering the heterogeneity of road networks in practice, selecting appropriate time-dependent functions improves the realism of the model. The distribution routes for RCdp5001 under Case 1 and Case 5 are shown in Figures 7 and 8.

In contrast, under static networks with different constant speeds, the differences are more pronounced. Compared with Case 5, the solutions for Case 6 and Case 7 involve 8 and 6 vehicles, respectively. Their travel distances deviate by 11.07% and −3.63%, travel times by 37.59% and −20%, carbon emission cost by 20.51% and −3.21%, and total costs by 34.82% and −7.73%. These results indicate that higher vehicle speeds substantially reduce both the total cost and the carbon emission.

In summary, optimal distribution solutions vary substantially under different road network conditions. By replacing static road representations with time-dependent networks, the proposed TDVRPSPDTW framework more accurately captures urban traffic dynamics and improves routing decisions. From a managerial perspective, the results highlight the importance of incorporating time-dependent traffic information to avoid underestimating operational costs and carbon emissions. From a policy perspective, the findings support investments in traffic monitoring and data-sharing infrastructure to facilitate low-carbon and congestion-aware urban logistics planning.

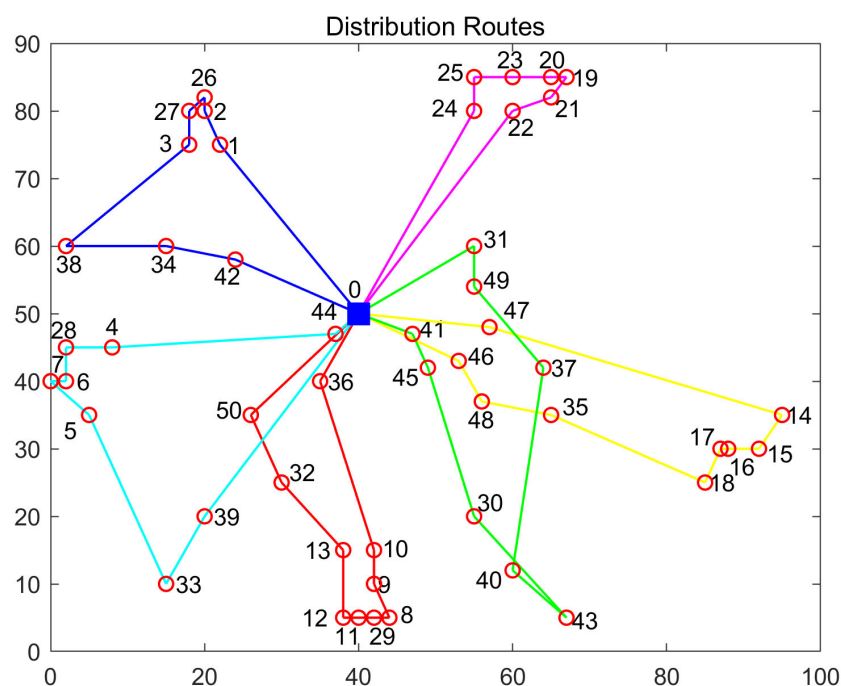


Figure 7. Distribution routes of RCdp5004 instance under Case 1 condition. The square represents the distribution center; the numbered circles represent the customers; different colors are the distribution paths of different distribution vehicles.

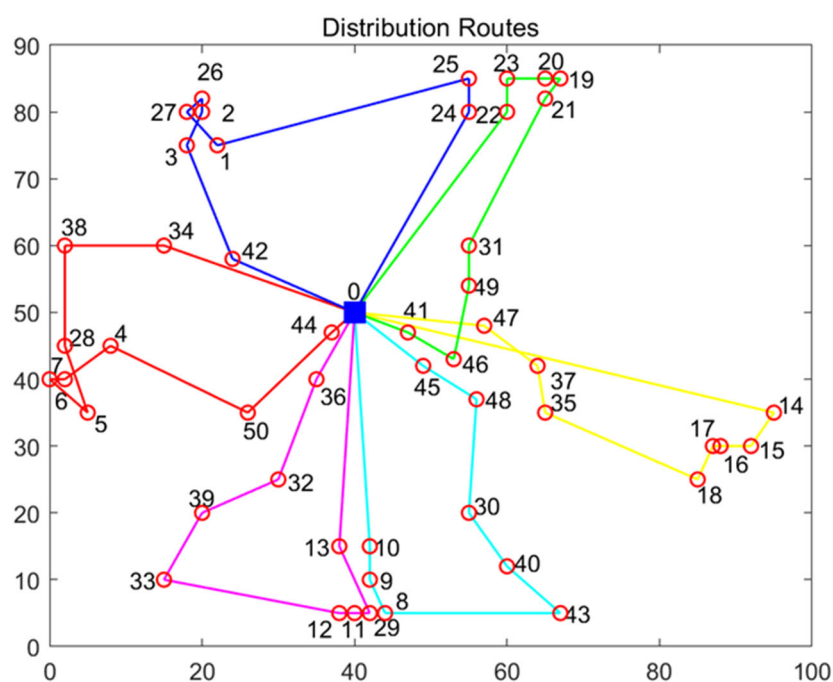


Figure 8. Distribution routes of RCdp5004 instance under Case 5 condition. The square represents the distribution center; the numbered circles represent the customers; different colors are the distribution paths of different distribution vehicles.

5.3.3. Influence of Time Window Characteristics

To examine the impact of time window characteristics on solution performance, the IACO algorithm is applied to solve the TDVRPSPDTW with strict time window constraints while keeping all other conditions unchanged, where vehicle services must be performed strictly within the specified customer time windows. While keeping all other

conditions and parameter settings unchanged, 10 independent runs are conducted, and the best solution is selected for comparison and analysis. Table 9 presents a comparison of results obtained under both soft and hard time windows. In the table, VT denotes the total violation time, i.e., the cumulative time that vehicles arrive earlier or later than customer time windows under the soft time window constraints, whereas WT represents the total waiting time incurred when vehicles arrive earlier than customer time windows under hard time window constraints.

Table 9. Solution results for TDVRPSPDTW instances under soft and hard time windows.

Instance	Soft Time Windows					Hard Time Windows				
	NV	TD	VT	CEC	TC	NV	TD	WT	CEC	TC
RCdp1001	2	280.56	107.72	6.88	1903.99	3	348.98	101.23	8.50	2442.12
RCdp1004	2	218.68	20.62	5.32	1566.71	2	218.68	20.62	5.32	1553.57
RCdp1007	2	246.27	69.74	6.03	1695.80	3	243.17	112.84	5.92	2055.25
RCdp2501	4	485.07	158.62	11.84	3504.43	6	546.37	180.75	13.34	4381.75
RCdp2504	4	515.41	51.02	12.65	3450.05	4	558.54	3.35	13.68	3632.26
RCdp2507	4	517.11	179.10	12.70	3596.30	5	562.54	33.88	13.73	3973.68
RCdp5001	6	745.81	768.78	18.25	6140.32	11	1087.6	354.58	26.56	8202.42
RCdp5004	6	699.57	86.61	17.11	4944.77	7	796.05	48.21	19.54	5574.29
RCdp5007	6	760.38	292.83	18.73	5475.33	8	945.53	33.82	22.63	6445.24

As shown in Table 9, compared with hard time window constraints, allowing early or delayed service through soft time windows with penalty costs can reduce the required fleet size and carbon emissions, thereby lowering the overall distribution cost. Specifically, the computational results indicate that the carbon emission cost can be reduced by up to 31.29%, while the total distribution cost can be reduced by up to 25.14% under soft time window settings.

From a managerial perspective, these results suggest that introducing moderate scheduling flexibility can improve fleet utilization and environmental performance without significantly compromising service quality. From a policy perspective, the findings indicate that flexible delivery time regulations, when supported by appropriate pricing or penalty mechanisms, can lead to measurable reductions in both operational costs and carbon emissions in urban logistics systems. Figures 9 and 10 further illustrate how different time window configurations result in distinct routing structures, for instance, RCdp5001.

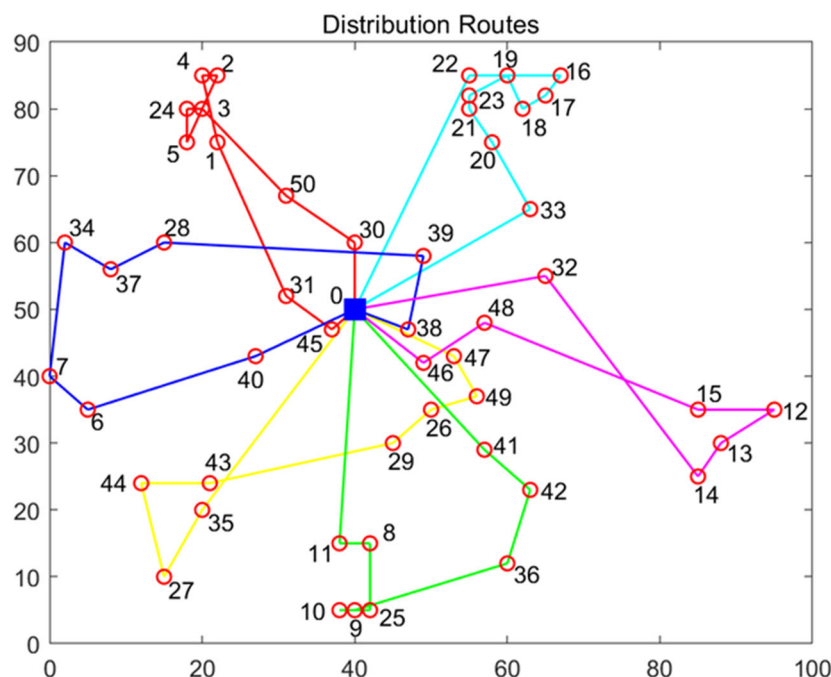


Figure 9. Distribution routes of RCdp5001 instance under soft time window. The square represents the distribution center; the numbered circles represent the customers; different colors are the distribution paths of different distribution vehicles.

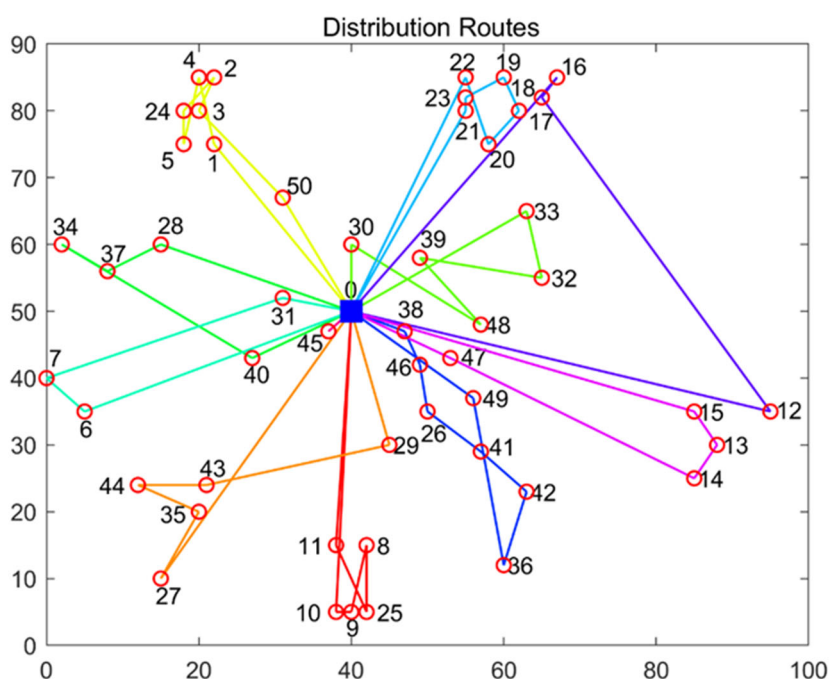


Figure 10. Distribution routes of RCdp5001 instance under hard time window. The square represents the distribution center; the numbered circles represent the customers; different colors are the distribution paths of different distribution vehicles.

5.3.4. Influence of Different Distribution Modes

To further evaluate the advantages of the simultaneous pickup–delivery strategy, additional experiments are conducted on the benchmark instance RCdp5004 under the conventional separate pickup–delivery mode. While keeping all other conditions, constraints, and parameter settings unchanged, 10 independent runs are performed, and the best

solution obtained is used for comparison. Table 10 reports the solution results under two distribution modes: simultaneous pickup–delivery and the “separate pickup + separate delivery”. The comparative results indicate that the separate pickup-and-delivery mode requires 12 vehicles, with a total carbon emission cost of 36.78 CNY and a total distribution cost of 9881.49 CNY. Compared with the simultaneous pickup–delivery mode, this represents an increase of 6 vehicles, a 114.96% higher carbon emission cost, and a 99.84% increase in total distribution cost. From a managerial perspective, these results indicate that adopting a simultaneous pickup–delivery strategy can substantially improve fleet utilization and operational efficiency, thereby avoiding redundant trips and excessive vehicle deployment. From a policy perspective, the findings suggest that encouraging integrated pickup–delivery operations through operational guidelines or incentive mechanisms can yield significant reductions in both logistics costs and carbon emissions, contributing to more efficient and sustainable urban freight systems.

Table 10. Solution results of the RCdp5004 instance in different distribution modes.

Distribution Modes		NV	Total NV	CEC	Total CEC	TC	Total TC
Separate pickup + separate delivery	Separate delivery	6	12	16.95	36.78	4927.61	9881.49
	Separate pickup	6		19.83		4953.88	
Simultaneous pickup–delivery		6	6	17.11	17.11	4944.77	4944.77

6. Conclusions and Future Research

6.1. Conclusions

To promote sustainable urban logistics development, this study extends the classical vehicle routing problem with time windows by formulating a time-dependent vehicle routing problem with simultaneous pickup–delivery and time windows (TDVRPSPDTW), aiming to minimize the total distribution cost. An improved ant colony optimization (IACO) algorithm is developed to efficiently solve the proposed model. Extensive numerical experiments and sensitivity analyses are conducted to validate the effectiveness of the model and algorithm. The main findings and contributions are summarized as follows:

1. First, a TDVRPSPDTW model is constructed by jointly considering time-dependent traffic conditions, simultaneous pickup–delivery demands, and carbon emission costs. It is a further deepening and expansion of the VRP. Based on the comprehensive emission model, it calculates the carbon emission cost of vehicles under time-dependent road networks, combines the time-dependent function to simulate the time-dependent characteristics of the road network and considers the simultaneous pickup–delivery demands of customers.
2. Second, the IACO incorporates several tailored strategies, including a savings-based heuristic for generating high-quality initial solutions, a modified state transition rule that considers distance-saving and time window deviation, and an embedded adaptive large neighborhood search mechanism to enhance exploration and prevent premature convergence. Experimental results on internationally recognized VRPSPDTW benchmark datasets show that the proposed IACO outperforms representative algorithms, such as CPLEX [22], DCS [58], MATE [59], CO-GA [17] and SFSSA [40], demonstrating its effectiveness and robustness in solving complex large-scale distribution problems.
3. Third, comparative experiments on constructed TDVRPSPDTW instances show that the proposed IACO yields lower optimal and average distribution costs than the ACO-LS algorithm. The average relative deviation of the optimal solutions is 4.23%, and the performance gap widens with increasing instance size—reaching up to 11.78% when the number of customers is 50.

4. Fourth, sensitivity analyses conducted on the constructed TDVRPSPDTW instances reveal the influence of time-dependent road networks, time window flexibility, and delivery modes on optimization performance. Compared with static networks, hard time windows, and separate pickup–delivery operations, the proposed model better reflects real urban logistics dynamics and improves routing efficiency. Incorporating time-dependent road networks helps alleviate peak-hour congestion effects; soft time windows reduce carbon emissions and distribution cost by 31.29% and 25.14%, respectively. Further, allowing simultaneous pickup–delivery demands lowers empty-load rates, reducing vehicle usage by 6, and cutting carbon emission and total cost by 53.48% and 50.29%, respectively.
5. Finally, beyond the methodological contributions, the proposed TDVRPSPDTW framework provides clear practical guidance for sustainable urban logistics operations. The model can be directly integrated into existing vehicle dispatching or logistics management systems by incorporating time-dependent traffic information, flexible time window settings, and simultaneous pickup–delivery strategies. From an operational perspective, logistics enterprises may apply the proposed approach in offline planning scenarios, such as daily or shift-based route scheduling, to improve fleet utilization, reduce empty-load rates, and achieve measurable reductions in distribution cost and carbon emissions. These implementation-oriented insights highlight the applicability of the proposed framework as a decision-support tool for low-carbon and congestion-aware urban logistics planning.

6.2. Limitations and Future Research

Despite the encouraging results, several limitations of this study should be acknowledged. To maintain model tractability, a number of simplifying assumptions are adopted, including homogeneous vehicle fleets, single-depot operations, and the exclusion of explicit cargo volume constraints, which may limit the direct applicability of the proposed framework to highly complex logistics systems. In addition, the proposed IACO incorporates an adaptive large neighborhood search mechanism to enhance diversification and convergence robustness, which inevitably introduces additional computational overhead, it remains within a practically acceptable range and allows the TDVRPSPDTW model to be solved effectively. Furthermore, the numerical experiments are conducted on synthetic benchmark instances, which may differ from real-world logistics environments characterized by stochastic demand and highly dynamic traffic conditions. Future research will focus on relaxing the above assumptions, incorporating real operational data, extending the model to heterogeneous fleets and multi-depot settings, as well as conducting a more systematic analysis of algorithmic parameters and adaptive mechanisms to further enhance robustness and scalability.

Author Contributions: Conceptualization, M.H.; methodology, M.H. and J.Z.; software, M.H., J.Z. and M.Y.; validation, M.H. and J.Z.; formal analysis, J.Z., X.Y. and M.Y.; investigation, M.H., J.Z. and X.M.; resources, M.H., X.H. and X.W.; data curation, M.H., J.Z. and M.Y.; writing—original draft, M.H. and J.Z.; writing—review and editing, M.H. and J.Z.; visualization, J.Z. and M.Y.; supervision, M.H. and X.H.; project administration, M.H.; funding acquisition, M.H., X.W. and J.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Postgraduate Research and Practice Innovation Program of Jiangsu Province (Grant number No. KYCX24_3984), Intelligent Policing Key Laboratory of Sichuan Province (Grant number No. ZNJW2026KFMS002, ZNJW2023KFMS004), and the Humanity and Social Science Youth Foundation of the Ministry of Education of China (Grant number No. 21YJCZH180).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Garg, V.; Niranjana, S.; Prybutok, V.; Pohlen, T.; Gligor, D. Drones in Last-Mile Delivery: A Systematic Review on Efficiency, Accessibility, and Sustainability. *Transp. Res. Part D Transp. Environ.* **2023**, *123*, 103831.
2. IEA. *Global Energy Review 2025*; IEA: Paris, France, 2025.
3. Guo, X.L.; Zhang, W.; Liu, B.B. Low-Carbon Routing for Cold-Chain Logistics Considering the Time-Dependent Effects of Traffic Congestion. *Transp. Res. Part D Transp. Environ.* **2022**, *113*, 103502.
4. Zhang, X.; Hao, Y.; Zhang, L.; Yuan, X. Application of Improved Genetic Algorithm to Vehicle Routing Problem Considering the Environmental Self-Regulation of the Freight Companies. *Expert. Syst. Appl.* **2025**, *274*, 127010.
5. Nan, Y. Multiobjective Optimization for Vehicle Routing Optimization Problem in Low-Carbon Intelligent Transportation. *IEEE Trans. Intell. Transp. Syst.* **2023**, *24*, 13161–13170.
6. Ren, C.X.; Lu, L.; Teng, J.; Yin, C.C.; Li, J.T.; Ji, H.W.; Wang, X.Q.; Fu, F.F.; Vincent, F.Y. Logistics Distribution Path Optimization Considering Carbon Emissions and Multifuel-Type Vehicles. *J. Adv. Transp.* **2025**, *2025*, 6668589.
7. Bajaj, A.S.; Dhodiya, J.M. Sustainable 4D TSP to Reduce Carbon Emission Using Reference Point MOQO Jaya Algorithm in Uncertain Environment. *J. Clean. Prod.* **2025**, *486*, 144487.
8. Dantzig, G.B.; Ramser, J.H. The Truck Dispatching Problem. *Manag. Sci.* **1959**, *6*, 80–91.
9. Clarke, G.; Wright, J.W. Scheduling of Vehicles from a Central Depot to a Number of Delivery Points. *Oper. Res.* **1964**, *12*, 568–581.
10. Solomon, M.M. Algorithms for the Vehicle Routing and Scheduling Problems with Time Window Constraints. *Oper. Res.* **1987**, *35*, 254–265.
11. Ichoua, S.; Gendreau, M.; Potvin, J. Vehicle Dispatching with Time-Dependent Travel Times. *Eur. J. Oper. Res.* **2003**, *144*, 379–396.
12. Fan, H.M.; Jun, Z.P.; Yu, C.; Xue, R.X. Dynamic Vehicle Routing Problem of Heterogeneous Fleets with Time-Dependent Networks. *Syst. Eng.-Theory Pract.* **2022**, *42*, 455–470.
13. Demir, E.; Bektaş, T.; Laporte, G. An Adaptive Large Neighborhood Search Heuristic for the Pollution-Routing Problem. *Eur. J. Oper. Res.* **2012**, *223*, 346–359.
14. Majidi, S.; Hosseini-Motlagh, S.; Ignatius, J. Adaptive Large Neighborhood Search Heuristic for Pollution-Routing Problem with Simultaneous Pickup and Delivery. *Soft Comput.* **2018**, *22*, 2851–2865.
15. Qin, G.; Tao, F.; Li, L.; Chen, Z. Optimization of the Simultaneous Pickup and Delivery Vehicle Routing Problem based on Carbon Tax. *Ind. Manag. Data Syst.* **2019**, *119*, 2055–2071.
16. Chen, J.; Liao, W.; Yu, C. Route Optimization for Cold Chain Logistics of Front Warehouses based on Traffic Congestion and Carbon Emission. *Comput. Ind. Eng.* **2021**, *161*, 107663.
17. Wang, H.F.; Chen, Y.Y. A Genetic Algorithm for the Simultaneous Delivery and Pickup Problems with Time Window. *Comput. Ind. Eng.* **2012**, *62*, 84–95.
18. Wang, H.; Chen, S.H.; Yin, X.Y.; Meng, L.X.; Wang, Z.W.; Wang, Z.F. A Hybrid Fuzzy C-Means Heuristic Approach for Two-Echelon Vehicle Routing with Simultaneous Pickup and Delivery of Multicommodity. *IEEE Trans. Fuzzy Syst.* **2025**, *33*, 218–230.
19. Xiao, Y.; Konak, A. A Simulating Annealing Algorithm to Solve the Green Vehicle Routing & Scheduling Problem with Hierarchical Objectives and Weighted Tardiness. *Appl. Soft Comput.* **2015**, *34*, 372–388.
20. Pan, B.B.; Zhang, Z.Z.; Lim, A. A Hybrid Algorithm for Time-Dependent Vehicle Routing Problem with Time Windows. *Comput. Oper. Res.* **2021**, *128*, 105193.
21. Kyriakakis, N.A.; Sevastopoulos, I.; Marinaki, M.; Marinakis, Y. A Hybrid Tabu Search-Variable Neighborhood Descent Algorithm for the Cumulative Capacitated Vehicle Routing Problem with Time Windows in Humanitarian Applications. *Comput. Ind. Eng.* **2022**, *164*, 107868.

22. Wang, C.; Mu, D.; Zhao, F.; Sutherland, J.W. A Parallel Simulated Annealing Method for the Vehicle Routing Problem with Simultaneous Pickup – Delivery and Time Windows. *Comput. Ind. Eng.* **2015**, *83*, 111–122.
23. Zhao, J.; Poon, M.; Tan, V.Y.F.; Zhang, Z. A Hybrid Genetic Search and Dynamic Programming-based Split Algorithm for the Multi-Trip Time-Dependent Vehicle Routing Problem. *Eur. J. Oper. Res.* **2024**, *317*, 921–935.
24. Wang, S.; Sun, W.; Huang, M. An Adaptive Large Neighborhood Search for the Multi-Depot Dynamic Vehicle Routing Problem with Time Windows. *Comput. Ind. Eng.* **2024**, *191*, 110122.
25. Srivastava, G.; Singh, A.; Mallipeddi, R. NSGA-II with Objective-Specific Variation Operators for Multiobjective Vehicle Routing Problem with Time Windows. *Expert. Syst. Appl.* **2021**, *176*, 114779.
26. Cai, Y.; Cheng, M.; Zhou, Y.; Liu, P.; Guo, J. A Hybrid Evolutionary Multitask Algorithm for the Multiobjective Vehicle Routing Problem with Time Windows. *Inform. Sci.* **2022**, *612*, 168–187.
27. Qi, R.; Li, J.; Wang, J.; Jin, H.; Han, Y. QMOEA: A Q-Learning-based Multiobjective Evolutionary Algorithm for Solving Time-Dependent Green Vehicle Routing Problems with Time Windows. *Inform. Sci.* **2022**, *608*, 178–201.
28. Wang, Y.; Ran, L.; Guan, X.; Fan, J.; Sun, Y.; Wang, H. Collaborative Multicenter Vehicle Routing Problem with Time Windows and Mixed Deliveries and Pickups. *Expert. Syst. Appl.* **2022**, *197*, 116690.
29. Nyako, S.I.; Tayachi, D.; Abdelaziz, F.B. Machine Learning Multi-Objective Optimization for Time-Dependent Green Vehicle Routing Problem. *Energ. Econ.* **2025**, *148*, 108628.
30. Li, Y.; Soleimani, H.; Zohal, M. An Improved Ant Colony Optimization Algorithm for the Multi-Depot Green Vehicle Routing Problem with Multiple Objectives. *J. Clean. Prod.* **2019**, *227*, 1161–1172.
31. Frey, C.M.M.; Jungwirth, A.; Frey, M.; Kolisch, R. The Vehicle Routing Problem with Time Windows and Flexible Delivery Locations. *Eur. J. Oper. Res.* **2023**, *308*, 1142–1159.
32. Liu, J.; Tong, L.; Xia, X. A Genetic Algorithm for Vehicle Routing Problems with Time Windows based on Cluster of Geographic Positions and Time Windows. *Appl. Soft Comput.* **2025**, *169*, 112593.
33. Lai, C.; Chiu, C.; Chen, T. A Nondominated Sorting Simplified Swarm Optimization with Local Search Mechanisms for Multi-Objective Vehicle Routing Problems with Time Windows. *Appl. Soft Comput.* **2025**, *174*, 112989.
34. Wu, Q.; Xia, X.; Song, H.; Zeng, H.; Xu, X.; Zhang, Y.; Yu, F.; Wu, H. A Neighborhood Comprehensive Learning Particle Swarm Optimization for the Vehicle Routing Problem with Time Windows. *Swarm Evol. Comput.* **2024**, *84*, 101425.
35. Maroof, A.; Ayvaz, B.; Naeem, K. Logistics Optimization Using Hybrid Genetic Algorithm (HGA): A Solution to the Vehicle Routing Problem with Time Windows (VRPTW). *IEEE Access* **2024**, *12*, 36974–36989.
36. Gmira, M.; Gendreau, M.; Lodi, A.; Potvin, J. Tabu Search for the Time-Dependent Vehicle Routing Problem with Time Windows on a Road Network. *Eur. J. Oper. Res.* **2021**, *288*, 129–140.
37. Wu, Y.; Ma, Z. Time-Dependent Production-Delivery Problem with Time Windows for Perishable Foods. *Syst. Eng.-Theory Pract.* **2017**, *37*, 172–181.
38. Ma, H.; Yang, T. 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* **2025**, *14*, 2375.
39. Pasupuleti, V.; Bharadwaj, T.; Chandra, S.K.; Malisetty, S. Enhancing Supply Chain Agility and Sustainability through Machine Learning: Optimization Techniques for Logistics and Inventory Management. *Logistics* **2024**, *8*, 73.
40. Liu, F.; Wang, L.; Gui, M.; Zhang, Y.; Lan, Y.; Lai, C.; Zhu, B. A Hybrid Heuristic Algorithm for Urban Distribution with Simultaneous Pickup-Delivery and Time Window. *J. Heuristics* **2023**, *29*, 269–311.
41. Little, J.D.C.; Murty, K.G.; Sweeney, D.W.; Karel, C. An Algorithm for the Traveling Salesman Problem. *Oper. Res.* **1963**, *11*, 972–989.
42. Andres Figliozzi, M. The Time Dependent Vehicle Routing Problem with Time Windows: Benchmark Problems, an Efficient Solution Algorithm, and Solution Characteristics. *Transp. Res. Part E Logist. Transp. Rev.* **2012**, *48*, 616–636.
43. Hof, J.; Schneider, M. An Adaptive Large Neighborhood Search with Path Relinking for a Class of Vehicle-Routing Problems with Simultaneous Pickup and Delivery. *Networks* **2019**, *74*, 207–250.
44. Shi, Y.; Zhou, Y.J.; Boudouh, T.; Grunder, O. A Lexicographic-based Two-Stage Algorithm for Vehicle Routing Problem with Simultaneous Pickup-Delivery and Time Window. *Eng. Appl. Artif. Intel.* **2020**, *95*, 103901.
45. Wu, H.; Gao, Y. An Ant Colony Optimization based on Local Search for the Vehicle Routing Problem with Simultaneous Pickup – Delivery and Time Window. *Appl. Soft Comput.* **2023**, *139*, 110203.
46. Chen, G.; Gao, J.; Chen, D. Research on Vehicle Routing Problem with Time Windows Based on Improved Genetic Algorithm and Ant Colony Algorithm. *Electronics* **2025**, *14*, 647.

47. Ren, T.; Luo, T.; Jia, B.; Yang, B.; Wang, L.; Xing, L. Improved Ant Colony Optimization for the Vehicle Routing Problem with Split Pickup and Split Delivery. *Swarm Evol. Comput.* **2023**, *77*, 101228.
48. Yu, V.F.; Jodiawan, P.; Gunawan, A. An Adaptive Large Neighborhood Search for the Green Mixed Fleet Vehicle Routing Problem with Realistic Energy Consumption and Partial Recharges. *Appl. Soft Comput.* **2021**, *105*, 107251.
49. Liu, W.; Zhou, Y.; Liu, W.; Qiu, J.; Xie, N.; Chang, X.; Chen, J. A Hybrid ACS-VTM Algorithm for the Vehicle Routing Problem with Simultaneous Delivery & Pickup and Real-Time Traffic Condition. *Comput. Ind. Eng.* **2021**, *162*, 107747.
50. Erdogdu, K.; Karabulut, K. Bi-Objective Green Vehicle Routing Problem. *Int. Trans. Oper. Res.* **2022**, *29*, 1602–1626.
51. Wen, M.Y.; Sun, W.; Yu, Y.; Tang, J.F.; Ikou, K.K. An Adaptive Large Neighborhood Search for the Larger-Scale Multi Depot Green Vehicle Routing Problem with Time Windows. *J. Clean. Prod.* **2022**, *374*, 133916.
52. Luo, H.; Dridi, M.; Grunder, O. A Branch-Price-and-Cut Algorithm for a Time-Dependent Green Vehicle Routing Problem with the Consideration of Traffic Congestion. *Comput. Ind. Eng.* **2023**, *177*, 109093.
53. Ahmed, Z.H.; Maleki, F.; Yousefikhoshbakht, M.; Haron, H. Solving the Vehicle Routing Problem with Time Windows Using Modified Football Game Algorithm. *Egypt. Inform. J.* **2023**, *24*, 100403.
54. Praxedes, R.; Bulhões, T.; Subramanian, A.; Uchoa, E. A Unified Exact Approach for a Broad Class of Vehicle Routing Problems with Simultaneous Pickup and Delivery. *Comput. Oper. Res.* **2024**, *162*, 106467.
55. Huang, S.; Huang, Y.; Blazquez, C.A.; Chen, C. Solving the Vehicle Routing Problem with Drone for Delivery Services Using an Ant Colony Optimization Algorithm. *Adv. Eng. Inform.* **2022**, *51*, 101536.
56. Wang, Y.; Wang, L.; Chen, G.; Cai, Z.; Zhou, Y.; Xing, L. An Improved Ant Colony Optimization Algorithm to the Periodic Vehicle Routing Problem with Time Window and Service Choice. *Swarm Evol. Comput.* **2020**, *55*, 100675.
57. Zhang, H.; Zhang, Q.; Ma, L.; Zhang, Z.; Liu, Y. A Hybrid Ant Colony Optimization Algorithm for a Multi-Objective Vehicle Routing Problem with Flexible Time Windows. *Inform. Sci.* **2019**, *490*, 166–190.
58. Wang, C.; Liu, C.; Mu, D.; Gao, Y. VRPSPDTW Problem Solving by Discrete Cuckoo Search. *Comput. Integr. Manuf. Syst.* **2018**, *24*, 570–582.
59. Liu, S.; Tang, K.; Yao, X. Memetic Search for Vehicle Routing with Simultaneous Pickup-Delivery and Time Windows. *Swarm Evol. Comput.* **2021**, *66*, 100927.
60. Liu, Z.; Wang, W.; He, J.; Zhang, J.; Wang, J.; Li, S.; Sun, Y.; Ren, X. A New Hybrid Algorithm for Vehicle Routing Optimization. *Sustainability* **2023**, *15*, 10982.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.