symmetry

MDPI

Article

Stage-Specific Multi-Objective Five-Element Cycle Optimization Algorithm in Green Vehicle-Routing Problem with Symmetric Distance Matrix: Balancing Carbon Emissions and Customer Satisfaction

Yue Xiang [1], Jingjing Guo [2], Zhengyan Mao [3], Chao Jiang [4] and Mandan Liu [1,*]

[1] Key Laboratory of Smart Manufacturing in Energy Chemical Process, Ministry of Education, East China University of Science and Technology, Shanghai 200237, China; yue.xiang@mail.ecust.edu.cn
[2] Department of Aerospace Science and Technology, Space Engineering University, Beijing 101416, China; jingjing.guo@mail.ecust.edu.cn
[3] Shanghai Key Laboratory of Computer Software Testing & Evaluating, Shanghai Development Center of Computer Software Technology, Shanghai 201112, China; maozy@sscenter.sh.cn
[4] Faculty of Information Technology, Beijing Key Laboratory of Computational Intelligence and Intelligent System, Engineering Research Center of Digital Community, Ministry of Education, Beijing Artificial Intelligence Institute and Beijing Laboratory for Intelligent Environmental Protection, Beijing University of Technology, Beijing 100124, China; chaojiang@mail.ecust.edu.cn
* Correspondence: liumandan@ecust.edu.cn

Abstract: This study presents a bi-objective optimization model for the Green Vehicle-Routing Problem in cold chain logistics, with a focus on symmetric distance matrices, aiming to minimize total costs, including carbon emissions, while maximizing customer satisfaction. To address this complex challenge, we developed a Stage-Specific Multi-Objective Five-Element Cycle Optimization algorithm (MOFECO-SS), which dynamically adjusts optimization strategies across different stages of the process, thereby enhancing overall efficiency. Extensive comparative analyses with existing algorithms demonstrate that MOFECO-SS consistently outperforms in solving the multi-objective optimization model, particularly in reducing total costs and carbon emissions while maintaining high levels of customer satisfaction. The symmetric nature of the distance matrix further aids in achieving balanced and optimized route planning. The results highlight that MOFECO-SS offers decision-makers flexible route planning options that balance cost efficiency with environmental sustainability, ultimately improving the effectiveness of cold chain logistics operations.

Keywords: vehicle routing problem; cold chain logistics; customer satisfaction; carbon emission; multi-objective five-element cycle optimization; exploration and exploitation

Citation: Xiang, Y.; Guo, J.; Mao, Z.; Jiang, C.; Liu, M. Stage-Specific Multi-Objective Five-Element Cycle Optimization Algorithm in Green Vehicle-Routing Problem with Symmetric Distance Matrix: Balancing Carbon Emissions and Customer Satisfaction. *Symmetry* **2024**, *16*, 1305. https://doi.org/10.3390/sym16101305

Academic Editor: Alexander Zaslavski

Received: 6 September 2024
Revised: 28 September 2024
Accepted: 1 October 2024
Published: 3 October 2024

1. Introduction

In recent years, as consumer demand for fresh, high-quality food has increased and attention to pharmaceutical safety has grown, the global cold chain logistics (CCL) industry has rapidly developed, becoming a key component of logistics and supply chain management [1,2]. However, the need for precise temperature control in cold chain logistics has led to higher energy consumption and carbon emissions, exacerbating environmental concerns and regulatory pressures [3]. Against this backdrop, the importance of green logistics has rapidly risen, giving rise to the Green Vehicle-Routing Problem (GVRP), which aims to minimize environmental impact while optimizing traditional logistics goals such as cost efficiency and customer satisfaction.

The GVRP has been extensively studied from two perspectives: model development and algorithmic solutions. Traditionally, GVRP models focus on minimizing total costs, which are particularly relevant in cold chain logistics due to the inclusion of additional factors such as goods damage costs and refrigeration costs. However, many studies have

simplified these models by neglecting certain key costs. For example, Zhang and Chen [4] did not consider goods damage costs, Osvald and Stirn [5] included goods damage costs but ignored refrigeration costs, and Wang et al. [6] only considered goods damage costs during unloading. To address these limitations, more comprehensive models have been proposed, such as the one by Wang et al. [7], which accounts for time penalty costs, refrigeration costs, cargo damage costs, fixed costs, and transportation costs within a time window-constrained optimization model.

As environmental protection becomes increasingly important, carbon emissions in cold chain logistics have gained significant attention [8]. Cold chain logistics systems consume more energy than their non-cold chain counterparts due to the need for refrigeration and temperature control, leading to higher carbon emissions from transportation, refrigeration equipment, and storage facilities [9]. This has prompted scholars to develop green logistics models that incorporate carbon emissions into the objective function, aiming to reduce the carbon footprint of logistics operations [10]. Recent studies have introduced low-carbon cold chain logistics models that minimize both distribution costs and carbon emissions [11–14], while other work has investigated the mechanisms behind carbon emissions for refrigerated trucks under different transportation conditions [15].

Customer satisfaction is also an increasingly important factor in cold chain logistics, particularly for businesses that depend on repeat customers and positive customer feedback [16]. Maintaining high customer satisfaction is critical to ensuring product quality, particularly for temperature-sensitive goods such as food and pharmaceuticals. Previous studies have explored the integration of customer satisfaction into logistics optimization models [17,18], but often overlook carbon emissions, failing to provide a holistic solution to the Green Vehicle-Routing Problem.

Despite the advancements in the field, there remains a lack of comprehensive models that simultaneously consider total costs, carbon emissions, and customer satisfaction. To address this gap, this paper proposes a bi-objective Green Vehicle-Routing Problem (BIGVRP) model. The model aims to minimize total costs, including carbon emissions, while maximizing customer satisfaction. By considering both operational efficiency and environmental sustainability, the model offers a more comprehensive framework for solving real-world logistics problems compared with traditional single-objective models, which typically focus only on minimizing costs.

In terms of algorithms used to solve GVRP models, both exact algorithms and heuristic methods have been applied. Exact algorithms, such as Lagrangian relaxation methods and integer programming, can achieve high accuracy in small-scale problems but are computationally expensive and impractical for large-scale logistics networks [19,20]. Decomposition-based algorithms have been shown to be highly sensitive to parameter settings, further limiting their practical application [15].

Heuristic algorithms, such as genetic algorithms [21], ant colony algorithms [22], particle swarm optimization (PSO), tabu search (TS) [23], artificial fish swarm algorithms [24], and the Clarke and Wright saving algorithm [25], have been favored for their ability to provide near-optimal solutions in large and complex scenarios. These algorithms can produce solutions in a reasonable amount of time, making them more practical for real-world logistics applications.

Building on this foundation, this study introduces the Stage-Specific Multi-Objective Five-Element Cycle Optimization (MOFECO-SS) algorithm, a significant innovation designed to tackle the BIGVRP. Unlike traditional algorithms, MOFECO-SS adopts a Stage-Specific approach, applying different evolutionary strategies at each stage of the optimization process to dynamically balance exploration and exploitation. This approach enables the algorithm to adapt to the changing demands of the problem as the optimization progresses, improving both convergence and solution diversity. MOFECO-SS is capable of generating high-quality Pareto front solutions that provide decision-makers with multiple delivery route options to balance environmental impact and operational goals.

Furthermore, the application of the Ideal Point Method to select a compromise solution from the Pareto front represents an additional contribution of this work. The Ideal Point

Method allows logistics managers to choose solutions that best align with their strategic priorities while accounting for trade-offs between conflicting objectives such as cost and customer satisfaction.

The remaining parts of the paper are organized as follows: Section 2 introduces the BIGVRP model proposed in this paper; Section 3 elucidates the principles and implementation process of the MOFECO-SS algorithm utilized herein in this study; Section 4 presents the experimental results and analysis; and Section 5 discusses the conclusions of the paper and suggests directions for future work.

2. Modeling Green Vehicle-Routing Problem

2.1. Problem Description

This paper targets the optimization problem within the BIGVRP, focusing on designing optimal routes for refrigerated trucks. The scenario involves all trucks originating from a distribution center, sequentially serving customer points, and then returning to the distribution center. Each customer point is assigned to a single truck, but each truck may serve multiple customer points. The central challenge is to decide which customer points each truck should serve and in what order, aiming to minimize total costs while maximizing customer satisfaction. Therefore, the primary goal is to establish efficient driving routes that start and end at the distribution center, striking a balance between cost-efficiency and meeting customer expectations. Figure 1 illustrates the scenario involving a distribution center and several customer points within a CCL network.

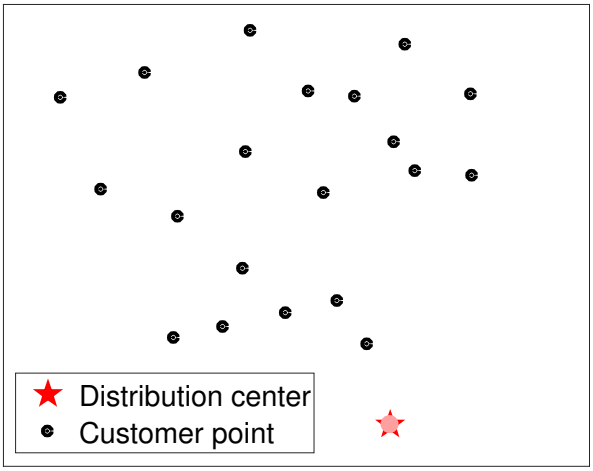


Figure 1. Distribution map of customer points and distribution center.

2.2. Problem Assumption

When studying GVRP models, it is common to make certain assumptions to simplify the complexity of the problem and focus on key variables. These assumptions are commonly applied in logistics optimization and have been adopted in a wide range of studies, such as [11,13,26], to create a tractable model that still closely represents real-world scenarios, providing a basis for comparison and validation of our results.

The model in this paper is developed for a real-world case involving the use of a company's own transport fleet for goods delivery. The following assumptions are made, which are crucial for building the model and conducting effective analysis:

- Uniform type of refrigerated trucks: It is assumed that all refrigerated trucks are of the same model, with identical load capacity and refrigeration performance. This assumption simplifies variability issues related to truck performance, a common practice in logistics optimization studies [13,26].
- Constant external and refrigeration temperatures: It is assumed that the external environmental temperature and the internal refrigeration temperature remain constant

throughout the entire delivery process. This assumption helps avoid the need for complex temperature control strategies and is consistent with previous studies such as [11].

- Clear roads: It is assumed that all roads are passable, and there are no traffic restrictions or road closures. Although this assumption simplifies traffic considerations, it allows us to focus on the core optimization of routing, a practice validated in earlier works [11,26].

The goal of these assumptions is to highlight the effectiveness of the proposed MOFECO-SS algorithm in addressing the core optimization challenge while minimizing the impact of unrelated variables. Similar simplifications have been shown to be effective in several studies. However, as discussed in related literature, future research can relax these assumptions to incorporate more complex and dynamic real-world factors, such as real-time traffic conditions, variable vehicle loads, and asymmetric distance matrices, making the model even more applicable to real-world scenarios.

2.3. Mathematical Model

BIGVRP proposed in this paper includes two objective functions, i.e., total costs and customer satisfaction.

2.3.1. Total Costs

The total cost refers to all the expenses incurred during the process of delivering goods to customer points, encompassing various handling fees throughout the entire transportation process. The calculation of total cost involves multiple stages and activities, each with its own unique characteristics and calculation methods. The total cost proposed in this paper consists of the following six types of costs:

- Fixed costs
  The fixed costs (C_f) refer to the expenses that remain constant regardless of the volume or frequency of deliveries. These costs typically involve the initial investment and ongoing operational expenses related to maintaining the necessary infrastructure and equipment for chilled or frozen product transportation. Such as the cost of hiring a driver, the cost of buying or leasing the refrigerated truck, the cost of tolls in the transportation process, the regular maintenance fee that may be needed for the refrigerated vehicle, and so on. In this paper, it is simplified to the fixed cost of each refrigerated truck, which is known and proportional to the number of refrigerated trucks needed in the transportation process. Therefore, the C_f can be expressed by the following formula:
  $$C_f = c_f K \tag{1}$$
  where the following is true:
  - c_f represents the fixed cost of each refrigerated truck.
  - K is the number of refrigerated trucks.
- Transportation costs
  Transportation costs (C_t) refer to the expenses incurred during the transportation of goods from the distribution center to customer points and back. While in real-world scenarios, transportation costs are influenced by various factors such as fuel consumption, vehicle load, road conditions, and traffic, in this paper, they are simplified to be only positively related to the transportation distance. This assumption is commonly used in logistics optimization models to reduce computational complexity and focus on the core routing problem, as demonstrated in studies such as [11,26]. The simplification allows us to validate the algorithm efficiently, while future research will incorporate more complex cost structures that account for dynamic traffic conditions and vehicle load.
  In this paper, the transportation distance for each refrigerated truck refers to the total distance it travels from the distribution center, completes all deliveries to assigned

customer points and returns to the distribution center. The transportation cost (C_t) is calculated using the following equation:

$$C_t = c_t \sum_{k=1}^{K} \left(d_{0,x_1^k} + \sum_{i=1}^{N_k-1} d_{x_i^k, x_{i+1}^k} + d_{x_{N_k}^k, 0} \right) \tag{2}$$

where the following is true:

- c_t is the transportation costs per unit distance.
- N_k is the number of customer points that the kth refrigerated truck is responsible for distribution.
- x_i^k is the ith customer point served by the kth refrigerated truck, $(i = 1, 2, 3, \cdots, N_k)$,
- $d_{x_i^k, x_{i+1}^k}$ denotes the distance from the ith customer point to the $(i+1)$th customer point of the kth refrigerated truck. This distance is assumed to be symmetric, i.e., $d_{x_i^k, x_{i+1}^k} = d_{x_{i+1}^k, x_i^k}$.
- d_{0,x_1^k} is the distance of the kth refrigerated truck from the distribution center (0) to the first customer point it serves.

While the current model simplifies transportation costs to be directly proportional to distance, future work will expand this to include real-time traffic conditions, fuel consumption, and load-dependent costs, making the model more reflective of real-world transportation scenarios. This will further enhance the applicability of the model in dynamic and complex logistics environments.

- Cargo damage costs

The costs of cargo damage (C_d) refers to the expenses incurred due to the loss or deterioration of the cargo during the transportation process. This is a significant cost component in GVRP, as it directly impacts the profitability and efficiency of the distribution operation.

Cargo damage can occur for various reasons in CCL. One primary cause is temperature fluctuations, which can lead to product spoilage or degradation. If the temperature within the refrigerated truck or storage facility is not maintained at the required level, the quality of the chilled or frozen products can be compromised. When the cargo is distributed, temperature fluctuations will be caused during transportation and during unloading.

Therefore, the cargo damage costs in the CCL mainly come from two parts: one is the cargo damage in the transportation process (C_{d_1}), and the other is the cargo damage during the unloading process (C_{d_2}).

They are expressed as follows:

$$C_d = C_{d_1} + C_{d_2} \tag{3}$$

C_{d_1} primarily depends on the time required to transport the cargo from the distribution center to the customer locations, as shown in the following equation:

$$C_{d_1} = c_d \sum_{k=1}^{K} \sum_{i=1}^{N_k} q_{x_i^k} \left(1 - e^{-\varepsilon_1 \left(t_{x_i^k} - t_0 \right)} \right) \tag{4}$$

where the following is true:

- c_d is the unit price of cargo.
- $q_{x_i^k}$ is the demand of the customer point x_i^k.
- ε_1 is the damage coefficient of cargo during transportation.
- $t_{x_i^k}$ indicates the time at which the kth refrigerated truck reaches the customer location x_i^k.
- t_0 represents the time at which the refrigerated trucks depart from the distribution center.

In fact, there are two parts of cargo damage in the unloading process. One part is the cargo damage that needs to be unloaded to the customer points in the unloading process. Since the cargo damage of this part is relatively small and it's too complicated to calculate, we take an approximation and ignore it. The other part is the remaining cargo quantity on the truck after the unloading process because the door of the refrigerated truck is always open, which causes damage to the cargo in the truck. In conclusion, the C_{d_2} are calculated by the following formula:

$$C_{d_2} = c_d \sum_{k=1}^{K} \sum_{i=1}^{N_k-1} Q_{x_i^k, x_{i+1}^k} \left(1 - e^{-\varepsilon_2 t^s_{x_i^k}}\right) \tag{5}$$

where the following is true:

- $Q_{x_i^k, x_{i+1}^k}$ is the load of the kth refrigerated truck in the process of driving from the customer point x_i^k to the customer point x_{i+1}^k.
- ε_2 is the damage coefficient of cargo during unloading.
- $t^s_{x_i^k}$ is the service time of the kth refrigerated truck at the customer point x_i^k.

- Refrigeration costs
  The refrigeration costs (C_r) refer to the expenses incurred in maintaining the required low temperatures throughout the transportation process to ensure the freshness and quality of the cargo. The C_r are mainly related to the time spent by all refrigerated trucks in the entire transportation process, which includes transportation time and unloading time and can be represented by the equation below:

$$C_r = c_{r_1} \sum_{k=1}^{K} \left(\frac{d_{0,x_1^k}}{v_{0,x_1^k}} + \sum_{i=1}^{N_k-1} \frac{d_{x_i^k, x_{i+1}^k}}{v_{x_i^k, x_{i+1}^k}} \right) + c_{r_2} \sum_{k=1}^{K} \sum_{i=1}^{N_k} t^s_{x_i^k} \tag{6}$$

  where the following is true:

  - c_{r_1} is the cost of refrigeration equipment per unit time during transportation.
  - c_{r_2} is the cost of refrigeration equipment per unit time during unloading.
  - v_{0,x_1^k} is the speed of the kth refrigerated truck when it leaves the distribution center.
  - $v_{x_i^k, x_{i+1}^k}$ is the speed of the kth refrigerated truck in the process of driving from the customer point x_i^k to the customer point x_{i+1}^k.
  - N represents the total number of customer points.

- Time penalty costs
  The time penalty costs (C_p) refer to the additional expenses incurred when the refrigerated trucks arrive at the customer points earlier or later than the expected time windows. These costs arise due to the inconvenience and potential loss caused by deviations from the expected delivery schedule.
  When refrigerated trucks arrive earlier than the expected time window, it may cause inconvenience to the recipient, as they may not be ready to receive the cargo. Conversely, late arrivals can lead to delays in the retail operations, causing potential sales losses. As a result, C_p is composed of costs incurred in advance and costs incurred due to delays. This can be represented by the following formula:

$$C_p = c_{p_1} \sum_{k=1}^{K} \sum_{i=1}^{N_k} \max\left(ET_{x_i^k} - t_{x_i^k}, 0\right) + c_{p_2} \sum_{k=1}^{K} \sum_{i=1}^{N_k} \max\left(t_{x_i^k} - LT_{x_i^k}, 0\right) \tag{7}$$

  where the following is true:

  - c_{p_1} is punishment cost per time due to the early arrival.
  - c_{p_2} is punishment cost per time due to the late arrival.
  - $\left[ET_{x_i^k}, LT_{x_i^k}\right]$ is the expected time window for customer point x_i^k.

- The carbon emissions costs
  In BIGVRP, carbon emission costs (C_e) are mainly related to total carbon emission, carbon quota, and carbon unit price:

$$C_e = c_e(EM - Q_c) \tag{8}$$

  where the following is true:
  - c_e is the carbon cost unit price.
  - EM represents the total carbon emissions.
  - Q_c is carbon emissions quotas.

  In a similar manner, carbon emissions account for both those generated during transportation (EM_t) and those produced during the unloading process (EM_r), as represented by the equation below:

$$EM = EM_t + EM_r \tag{9}$$

  Carbon emissions during transportation EM_t are mainly related to fuel consumption (F_c), as shown in the following equation:

$$EM_t = \eta F_c \tag{10}$$

  where η is the coefficient values of the carbon emissions.
  According to reference [9], F_c is related to the travel distance and load of the trucks, and F_c per unit distance is calculated as follows:

$$\rho(X) = \rho_0 + \frac{\rho^* - \rho_0}{Q} X \tag{11}$$

  where the following is true:
  - X is the load.
  - Q is the rated load of the refrigerated truck.
  - ρ_0 is the fuel consumption per unit distance when the truck is unloaded.
  - ρ^* is the fuel consumption per unit distance when the truck is fully loaded.

  Therefore, the EM_t consists of three parts, like the cargo damage costs, as shown in the following equation:

$$EM_t = \eta \sum_{k=1}^{K} \left\{ \left(\rho_0 + \frac{\rho^* - \rho_0}{Q} Q_{0,x_1^k} \right) d_{0,x_1^k} + \sum_{i=1}^{N_k - 1} \left[\left(\rho_0 + \frac{\rho^* - \rho_0}{Q} Q_{x_i^k, x_{i+1}^k} \right) d_{x_i^k, x_{i+1}^k} \right] + \rho_0 d_{x_{N_k}^k, 0} \right\} \tag{12}$$

  where:
  - $Q_{x_i^k, x_{i+1}^k}$ denotes the load carried by the kth refrigerated truck as it moves from customer location x_i^k to customer location x_{i+1}^k.
  - Q_{0,x_1^k} is the load of the kth refrigerated truck when it leaves the distribution center.

  The carbon emissions (EM_r) during the refrigeration process are related to the amount of cargo and their refrigeration time, similar to the refrigeration costs, and are calculated using the following formula:

$$EM_r = \mu \left[\sum_{k=1}^{K} \left(Q_{0,x_1^k} \frac{d_{0,x_1^k}}{v_{0,x_1^k}} + \sum_{i=1}^{N_k - 1} Q_{x_i^k, x_{i+1}^k} \frac{d_{x_i^k, x_{i+1}^k}}{v_{x_i^k, x_{i+1}^k}} \right) + \sum_{k=1}^{K} \sum_{i=1}^{N_k - 1} t_{x_i^k}^s Q_{x_i^k, x_{i+1}^k} \right] \tag{13}$$

  where μ is carbon emissions from cooling per unit weight of cargo per unit time.

While the current model focuses on minimizing carbon emissions based on distance traveled, it does not take into account the impact of real-time traffic conditions, which can alter

fuel consumption and carbon emissions. Future work could incorporate real-time traffic data and dynamic route adjustments to more accurately reflect emissions in practical applications.

2.3.2. Customer Satisfaction

In the BIGVRP proposed in this paper, customer satisfaction is mainly related to the timeliness of delivery. The relationship between customer satisfaction and delivery time is shown in Figure 2.

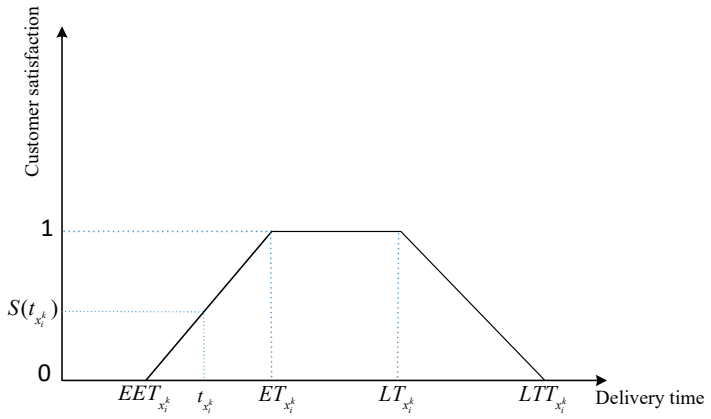


Figure 2. Relationship between customer satisfaction and delivery time.

In Figure 2, customer satisfaction $S(t_{x_i^k})$ at a specific customer point x_i^k in the cold chain logistics is related to the delivery time $t_{x_i^k}$. $\left[EET_{x_i^k}, LLT_{x_i^k}\right]$ is the acceptable time window for customer point x_i^k, and $\left[ET_{x_i^k}, LT_{x_i^k}\right]$ is the expected time window. According to the range of $t_{x_i^k}$, the value of $S(t_{x_i^k})$ is divided into the following four situations:

- Before $EET_{x_i^k}$ and after $LLT_{x_i^k}$:
  The satisfaction score is 0. This indicates complete dissatisfaction as the delivery time is either too early or too late, far outside the acceptable time window for the customer.
- Between $EET_{x_i^k}$ and $ET_{x_i^k}$:
  Satisfaction increases linearly from 0 to 1 as the delivery time approaches the expected time from the earliest time. This reflects increasing satisfaction as the delivery time nears the most desired point.
- Between $ET_{x_i^k}$ and $LT_{x_i^k}$:
  Satisfaction remains at its maximum of 1, indicating that delivery within this time frame meets the customer's expectations perfectly.
- Between $LT_{x_i^k}$ and $LLT_{x_i^k}$:
  Satisfaction decreases linearly from 1 back down to 0, indicating diminishing satisfaction as delivery becomes progressively later than the customer's latest expected time.

Certainly, the customer satisfaction function at each customer point can be represented mathematically in Equation (14). To compute the overall customer satisfaction for the entire delivery process, we take the weighted average of the satisfaction scores for all customer points, with the weights being the demand at each customer point, as shown in Equation (15).

$$S\left(t_{x_i^k}\right) = \begin{cases} \dfrac{t_{x_i^k} - EET_{x_i^k}}{ET_{x_i^k} - EET_{x_i^k}}, & \text{if } EET_{x_i^k} \le t_{x_i^k} < ET_{x_i^k} \\ 1, & \text{if } ET_{x_i^k} \le t_{x_i^k} \le LT_{x_i^k} \\ \dfrac{LLT_{x_i^k} - t_{x_i^k}}{LLT_{x_i^k} - LT_{x_i^k}}, & \text{if } LT_{x_i^k} < t_{x_i^k} \le LLT_{x_i^k} \\ 0. & \text{other} \end{cases} \tag{14}$$

$$S_{\text{total}} = \frac{\sum_{k=1}^{K} \sum_{i=1}^{N_k} S\left(t_{x_i^k}\right) q_{x_i^k}}{\sum_{k=1}^{K} \sum_{i=1}^{N_k} q_{x_i^k}} \tag{15}$$

To facilitate unified processing and convert the problem of maximizing customer satisfaction into a minimization problem, we introduce the concept of average customer dissatisfaction, calculated as follows:

$$f_2 = 1 - S_{\text{total}} \tag{16}$$

Equation (16) represents the objective function for minimizing customer dissatisfaction.

2.3.3. Bi-Objective Green Vehicle-Routing Problem Model

The BIGVRP in CCL proposed in this study includes two objective functions, namely minimizing total costs and minimizing customer dissatisfaction.

- Minimizing total costs:

$$\min f_1 = C_f + C_t + C_d + C_r + C_p + C_e \tag{17}$$

- Minimizing customer dissatisfaction:

$$\min f_2 = 1 - \frac{\sum_{k=1}^{K} \sum_{i=1}^{N_k} S\left(t_{x_i^k}\right) q_{x_i^k}}{\sum_{k=1}^{K} \sum_{i=1}^{N_k} q_{x_i^k}} \tag{18}$$

With these two objective functions, the BIGVRP model can simultaneously consider both costs and customer satisfaction, thereby optimizing the routes of BIGVRP in CCL. Equation (16) expresses customer dissatisfaction in a minimization form, making the handling of the optimization problem more consistent and convenient.

Subject to the following:

$$Q_{0,x_1^k} < Q \tag{19}$$

$$Q_{0,x_1^k} = \sum_{i=1}^{N_k} q_{x_i^k} \tag{20}$$

$$Q_{x_{N_k}^k,0} = 0 \tag{21}$$

$$\bigcup_{k=1}^{K} \{x_1^k, \cdots, x_{N_k}^k\} = X,\ X = \{x_1, x_2, \cdots, x_N\} \tag{22}$$

$$\sum_{k=1}^{K} N_k = N \tag{23}$$

$$x_i^k \neq x_j^{k'}, k \neq k' \tag{24}$$

where $x_i^k \in \{x_1^k, \cdots, x_{N_k}^k\}, \forall k \in \{1, 2, \cdots, K\}, x_j^{k'} \in \{x_1^{k'}, \cdots, x_{N_{k'}}^{k'}\}, \forall k' \in \{1, 2, \cdots, K\}$.

Equation (17) is to minimize the total cost of the whole transportation process, making the logistics operation as cost-effective as possible while maintaining the required service quality. Equation (18) is aimed at minimizing the average dissatisfaction among customers (ACDS), thereby enhancing customer satisfaction by ensuring deliveries meet both timing and condition expectations set by the customers. The formula inverts the measure of

satisfaction to frame it as a minimization problem, allowing for its integration into a multi-objective optimization framework.

In Equation (18), customer satisfaction is weighted by the quantity of goods they order, meaning customers with larger orders have a greater impact on the overall satisfaction score. This assumption is grounded in typical logistics operations, where larger orders often represent higher revenue or operational efficiency and thus receive higher priority in routing decisions. Conversely, customers with smaller orders may receive lower priority, which can affect their delivery timing or conditions and thus reduce their satisfaction. As such, this weighting system reflects the practical trade-offs faced in real-world distribution networks, where resources must be allocated efficiently to serve high-demand customers while still maintaining service levels for smaller-demand customers.

These two equations together are employed in a multi-objective optimization framework, where trade-offs between costs and customer satisfaction are analyzed. This allows for determining the best operational strategies that balance delivery efficiency and customer satisfaction across the distribution network.

Equation (19) ensures that no refrigerated truck exceeds its rated capacity. Equations (20) and (21) confirm that each truck begins and ends its journey at the distribution center after completing all assigned deliveries. Equations (22) through (24) stipulate that each customer is serviced by exactly one refrigerated truck, with Equations (22) and (23) ensuring that all customer locations are visited by a truck, and Equation (24) guaranteeing that each customer is visited by only one truck.

The core challenge of the BIGVRP in CCL is to find a balance between reducing overall costs and enhancing customer satisfaction. The optimal routes for minimizing costs may not align with the optimal routes for maximizing customer satisfaction. For example, a route that minimizes fuel consumption and toll costs may result in deliveries being made outside the preferred time windows for some customers, leading to lower satisfaction. Conversely, routes that maximize customer satisfaction by adhering strictly to preferred delivery times may incur higher transportation and refrigeration costs.

This trade-off is a critical aspect of our model, as it requires balancing competing objectives to find solutions that provide a satisfactory compromise between cost efficiency and customer satisfaction. The proposed MOFECO-SS is designed to effectively navigate this trade-off by dynamically adapting the optimization process through different stages, enhancing the overall efficacy of the search for optimal solutions.

3. Implementation of Stage-Specific Multi-Objective Five-Element Cycle Optimization Algorithm for BIGVRP

3.1. Five-Element Cycle Model

The five elements theory, a concept rooted in Chinese philosophy, is used to explain the formation of all things in the world. This theory focuses on the interactions between metal, wood, water, fire, and earth, which are governed by both generative and restrictive cycles. The generative cycle can be compared with the relationship between a mother and her child, where the child relies on the mother for nourishment and development. The order of generation proceeds as follows: wood creates fire, fire forms earth, earth generates metal, metal produces water, and water nourishes wood. In contrast, the restrictive cycle is similar to the relationship between a grandparent and a grandchild, where the grandparent imposes discipline. The sequence of restrictions is as follows: wood restricts earth, earth limits water, water controls fire, fire restricts metal, and metal constrains wood. Achieving harmony between generation and restriction is key to the balanced transformation of the five elements. Each element is influenced by the other four in various ways. For instance, wood produces fire and is generated by water, while it controls earth and is constrained by metal. This complex interaction ensures that nature maintains a dynamic equilibrium (as discussed in Reference [27]).

As depicted in Figure 3, the blue arrows in the outer cycle represent the generative process, while the red arrows in the inner cycle depict the restrictive process. The generative interactions can be understood through the parent–child relationship, where parents assist

in the growth of their children. On the other hand, restrictive interactions can be likened to the relationship between grandparents and grandchildren, where the older generation imposes limits on the younger one, creating a dynamic balance.

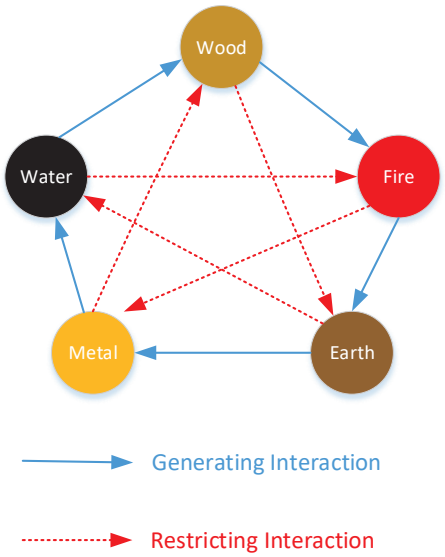


Figure 3. The generating and restricting interaction among five elements.

The Five-element Cycle Model (FECM) is constructed based on the Five-element theory [27], which extends the interactions of these five elements to a broader general case. Consider a dynamic system with L elements. At time g, the force exerted on element $\mathbf{x}_l(g)$ $(l = 1, 2, \cdots, L)$ by other elements in the cycle is defined as $F_l(g)$, and these forces depend on the respective masses of the elements. The mass of each element is denoted by $m_l(g)$. The FECM is formulated as follows:

$$\begin{cases} F_l(g) = \omega_{gp} \ln\left[\frac{m_{l-1}(g)}{m_l(g)}\right] - \omega_{rp} \ln\left[\frac{m_{l-2}(g)}{m_l(g)}\right] - \omega_{ga} \ln\left[\frac{m_i(k)}{m_{i+1}(g)}\right] - \omega_{ra} \ln\left[\frac{m_l(g)}{m_{l+2}(g)}\right] \\ m_l(g+1) = m_l(g) \cdot \frac{2}{1+\exp(-F_l(g))} \end{cases} \tag{25}$$

where $l = 1, 2, \ldots, L$, when $l = 1$, $l-1$ is replaced by L, when $l = 2$, $l-2$ is replaced by L, when $l = L$, $l+1$ is replaced by 1, when $l = L-1$, $l+2$ is replaced by 1, ω_{gp}, ω_{rp}, ω_{ga} and ω_{ra} are weight coefficients, in general, $\omega_{gp} = \omega_{rp} = \omega_{ga} = \omega_{ra} = 1$.

3.2. Multi-Objective Five-Element Cycle Algorithm

The Five-element Cycle Optimization Algorithm (FECO) [27] was derived from the FECM and developed as an iterative algorithm to solve optimization problems. In this framework, each element symbolizes a potential solution. All elements are organized into cycles, with each cycle consisting of a set number of elements. The value of the objective function for each element is treated as its mass, represented by m. The relationship between the mass m and the force F within the FECM is used to guide the update strategy for solutions. The optimal solution is found through iterative processes.

Building on FECO, the Multi-objective Five-element Cycle Optimization algorithm (MOFECO) was developed to handle Multi-objective Optimization Problems (MOPs) [28]. MOPs typically involve multiple objectives, and each element in MOFECO is subjected to several influencing forces, which introduces new techniques for evaluating and adjusting the elements.

3.3. Expression of Solution and Objective Function in MOFECO-SS

When the MOFECO-SS is applied to BIGVRP, each distribution plan is modeled as a solution within the algorithm, also referred to as an element in the FECM. We use $\mathbf{x}_{lr}(g)$ to denote the lth element in the rth cycle at time g.

$$\mathbf{x}_{lr}(g) = [x_{1,lr}(g),\ x_{2,lr}(g),\ \cdots,\ x_{N,lr}(g)] \tag{26}$$

where $l = 1, 2, \cdots, L$, $r = 1, 2, \cdots, R$. The force exerted on each element is calculated using the formula:

$$F_{lr,j}(g) = \ln\left[\frac{m_{(l-1)r,j}(g)}{m_{lr,j}(g)}\right] - \ln\left[\frac{m_{(i-2)r,j}(g)}{m_{lr,j}(g)}\right] - \ln\left[\frac{m_{lr,j}(g)}{m_{(l+1)r,j}(g)}\right] - \ln\left[\frac{m_{lr,j}(g)}{m_{(l+2)r,j}(g)}\right] \tag{27}$$

where j indicates the force corresponding to the jth objective function, with $j = 1, 2$ in this paper.

In MOFECO-SS, the value of each objective function is treated as the mass of each element. However, according to Equation (18), the second objective function may be 0, so we add a small value to each objective function to ensure that Equation (27) remains meaningful. Therefore,

$$m_{lr,j}(g) = f_j(\mathbf{x}_{rl}(g)) + \varepsilon \tag{28}$$

where ε is a small positive number; in this study, the value is set to 0.000001.

Based on the relationship among FECM, MOFECO-SS, and BIGVRP as shown in Table 1, MOFECO-SS is designed to solve the BIGVRP.

Table 1. Relationship among FECM, MOFECO-SS, and BIGVRP.

FECM	MOFECO-SS	BIGVRP
Elements $\mathbf{x}_l(g)(l = 1, 2, \cdots, L)$	Elements $\mathbf{x}(g)$ $(l = 1, 2, \cdots, L; r = 1, 2, \cdots, R)$	Solution (Route)$\mathbf{x}$
Mass of elements $m_l(g)$	Mass of elements $m_{lr,j}(g)$ $(l = 1, 2, \cdots, L; r = 1, 2, \cdots, R; j = 1, 2)$	Objective functions (Total costs f_1 and average dissatisfaction among customer f_2)
Force exerted on elements $F_l(g)$	Force exerted on elements $F_{lr,j}(g)$	Variables estimating the quality of solutions

3.4. Sorting Mechanism

The sorting mechanism plays a decisive role, particularly in the selection and updating of solutions. This mechanism involves ranking individuals to determine which will participate in the next phase of updates. Not only does this help the algorithm efficiently search the solution space, but by applying different updating strategies based on the ranking results, it further enhances the breadth and depth of the search. In MOFECO-SS, we have implemented two sorting mechanisms to rank the elements.

3.4.1. Force-Based Sorting Mechanism

The first mechanism arranges the elements in each cycle according to the forces applied to them, as illustrated in Figure 4, where red and blue distinguish the forces on each element from two objective functions, respectively. This helps illustrate the ranking process for each element based on both objectives. The process described in the diagram involves sorting elements within a cycle ($L = 5$) in the algorithm. The specific steps are as follows:

- Initial sorting based on forces:
  Initially, each element is ranked individually based on the forces associated with each of its objective functions. This means that if an element is influenced by multiple objectives, it will receive a separate ranking number for each objective.
- Summation of rankings for each objective:
  Subsequently, the ranking numbers for each element across all objective functions are aggregated to calculate a total. This step consolidates the individual rankings into

a comprehensive score, reflecting the overall performance of the element across all considered objectives.

- Final sorting based on accumulated scores:
  The total sum of rankings is then used to sort the elements once again. This final sorting based on the accumulated scores determines the final order of the elements within the cycle. This method ensures that elements that perform well across multiple objectives are given appropriate priority.

In Figure 4, taking element 5 as an example, each force acting on it is sorted individually, resulting in ranks of 1 and 3, respectively. Upon aggregating these ranks, element 5 attains the lowest cumulative rank within the cycle, thereby categorizing it as the element with relatively the poorest performance.

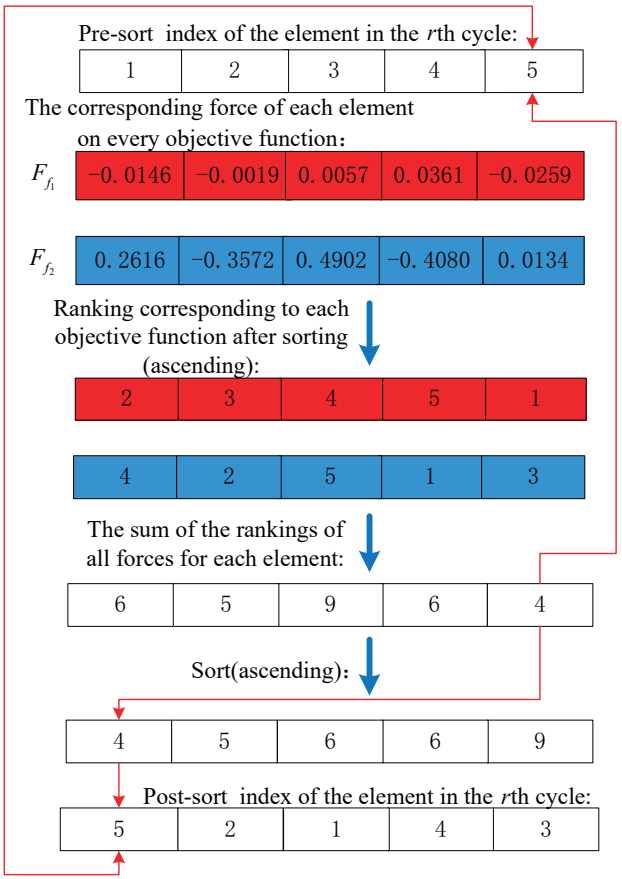


Figure 4. The arrangement of elements within each cycle.

This sorting method is crucial for the algorithm that needs to consider multiple evaluation criteria simultaneously. It ensures that elements that perform well across a broad range of objectives are recognized and appropriately prioritized, thereby optimizing the overall performance of the algorithm.

Utilizing this methodology, the elements in each cycle are systematically arranged according to their total scores. The element with the top ranking in each cycle, represented as $\mathbf{x}_{br}(g)$, represents the most optimal solution within the rth cycle as determined by the force-based sorting mechanism. This element exemplifies the highest efficacy in meeting the algorithm's objectives according to the calculated forces, establishing a benchmark for subsequent iterations and refinements within the cycle.

3.4.2. Fast Non-Dominated Sorting Mechanism

The second sorting mechanism implemented within this framework employs the well-established fast non-dominated sorting approach, as delineated by [29]. This method is strategically applied after each iteration cycle to rank the elements and to accurately identify all non-dominated solutions present within the population. In line with this methodology, the non-dominated solutions are represented by P^*, which signifies those solutions in the current population that have not been outperformed in any of the objectives, thereby reflecting the most efficient trade-offs among the considered criteria. Any solution within the P^* is denoted by $\mathbf{x}_{nd}(g)$.

3.5. Encoding and Decoding

In addressing BIGVRP, encoding and decoding are crucial steps. They involve transforming the actual delivery routes into a format that algorithms can process (encoding) and converting algorithm outputs back into executable delivery routes (decoding). This process allows algorithms to efficiently handle and optimize delivery routes.

The encoding method we use in this paper is permutation encoding. There are N customer points, K trucks, and the distribution center is represented by 0. As shown in Figure 5, assume that $N = 10$, $K = 3$.

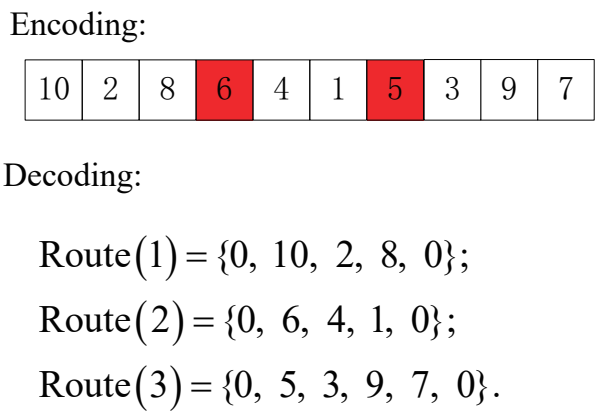


Figure 5. Encoding and decoding.

Each customer point is assigned a unique identifier, and each delivery route of the trucks is represented by a sequence of these identifiers along with the distribution center. The order of the sequence indicates the order in which these customer points are visited. For example, $Route(1) = \{0, 10, 2, 8, 0\}$ indicates that truck 1 departs from the distribution center, then serves customer point 10, and so forth, eventually returning to the distribution center after completing deliveries. In Figure 5, red blocks indicate that customer point 6 is served by truck 2, and customer point 5 is served by truck 3. The initial plan is determined based on the constraints of the refrigerated truck's rated load capacity Q. If truck 1 is also to deliver to customer point 6, it would result in an overload, so customer point 6 is assigned to truck 2 for service. This arrangement can lead to a problem where the demand from all remaining customer points served by the last truck may exceed its rated load capacity. In this case, to facilitate the elimination of infeasible solutions by the algorithm, a substantial number Q_p is added as a penalty term to the objective function. The pseudocode for encoding and decoding is shown as Algorithm 1.

Algorithm 1 Encoding and decoding

$Q_k = \emptyset$, Route(k)=$\emptyset$, $k = 1, 2, \cdots, K$
Set $k = 1$
for $i = 1 \rightarrow N$ **do**
  if $k < K$ **then**
    $Q_k = Q_k + q_{x_i}$
    if $Q_k \leq Q$ **then**
      Route(k) = {Route(k) x_i}
    else
      $Q_k = Q_k - q_{x_i}$
      $k \leftarrow k + 1$
      $Q_k = Q_k + q_{x_i}$
      Route(k) = {Route(k) x_i}
    end if
  else
    Route(k) = {Route(k) x_i}
    $Q_k = Q_k + q_{x_i}$
  end if
end for
if $Q_k > Q$ **then**
  $Q_p = 100,000,000 * (Q_k - Q)$
end if

3.6. Crossover and Mutation

In MOFECO-SS, crossover and mutation are two pivotal genetic operators used to generate new populations, explore the solution space, and enhance diversity within the population. These techniques mimic the genetic mechanisms of natural organisms, aiding the algorithm in finding potentially optimal solutions.

There are numerous methods for performing crossover and mutation, and selecting the most effective techniques is essential for enhancing the algorithm's search capabilities. Given the use of permutation encoding in this study, we have opted for the Partially Matched Crossover approach and the flip mutation method. Figures 6 and 7 illustrate the processes associated with these two operators, where the colored blocks represent the genes that are altered during the crossover and mutation processes.

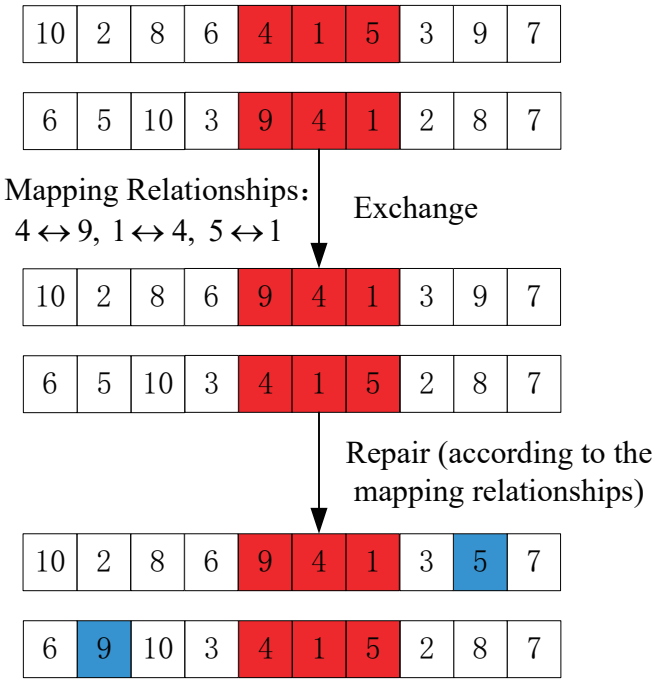


Figure 6. Partially Matched Crossover.

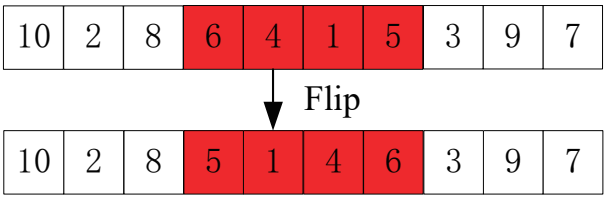


Figure 7. Flip mutation.

3.7. Stage-Specific Strategies of Evolution

As an iterative algorithm, MOFECO stands out for its dynamic approach to enhancing the quality of the population across multiple iterations, with the ultimate aim of identifying the optimal or near-optimal solutions. Based on this, we propose MOFECO-SS. The concept of "Stage-Specific" in our algorithm refers to the tailored strategies applied during different stages of the optimization process, enhancing the balance between exploration and exploitation to improve the solution quality and convergence speed. Implementing distinct evolutionary strategies tailored to specific stages of the iteration process is instrumental in augmenting both the efficiency of the algorithm and the quality of the resultant solutions. This strategic adaptation facilitates a crucial equilibrium between the exploration of the solution space and the exploitation of promising regions, thereby bolstering the capability of algorithms to circumvent local optima and accelerate convergence towards global optima.

In the MOFECO-SS, the evolutionary process is divided into three stages based on the termination condition, which is defined by the number of objective function evaluations. The algorithm concludes its iterations when the count of evaluations, denoted as c, reaches a predefined threshold c_{max}. The initial stage of evolution occurs when c is within the first 20% of c_{max}. The mid-stage takes place when c is between 20% and 80% of c_{max}. Finally, the last 20% of c_{max} marks the late stage of evolution.

3.7.1. Initial Stage: Intensifying Exploration

In the initial stage of the MOFECO-SS, the primary objective is to explore a broad range of the solution space. Given that the initial population is generated randomly, this stage inherently benefits from a high level of diversity among the elements, presenting an excellent opportunity for exploring new solution spaces. Building upon this foundation, we enhance exploration through crossover and mutation operators. For each element $\mathbf{x}_{lr}(g)$, select a random cycle r' distinct from its own cycle and choose the best element in the cycle, denoted by $\mathbf{x}_{br'}(g)$, $r \neq r'$. Then use the crossover operator to combine the $\mathbf{x}_{lr}(g)$ with the $\mathbf{x}_{br'}(g)$, generating a new element. Finally, apply a higher mutation rate p_{m_is} to mutate the new element, further broadening the solution space.

3.7.2. Mid-Stage: Balancing Exploration and Exploitation

During the mid-stage of the evolutionary process, which encompasses a significant portion of the total evaluations, the algorithm begins to balance its approach between further exploration and the exploitation of promising solutions. While new solutions are still explored, the emphasis gradually shifts towards refining the existing ones. This involves a more nuanced application of crossover and mutation operations, where the parameters are adjusted to foster incremental improvements rather than drastic changes. Here's a detailed look at how this progression is implemented:

Based on the sorting mechanism of force, elements within the cycle are arranged in order from worst to best, as shown in Figure 8. We utilize a parameter, denoted as p_{c_ms}, to divide the elements in each cycle into two groups: one representing relatively inferior elements (denoted as $L * p_{c\_ms}$), and the other representing relatively superior elements (notated as $L * (1 - p_{c\_ms})$. The different colors in the figure are used to distinguish these two groups, with red indicating the worst element and blue representing the best element. We employ elements from these two groups to perform crossover operations with the $\mathbf{x}_{nd}(g)$ and $\mathbf{x}_{br'}(g)$.

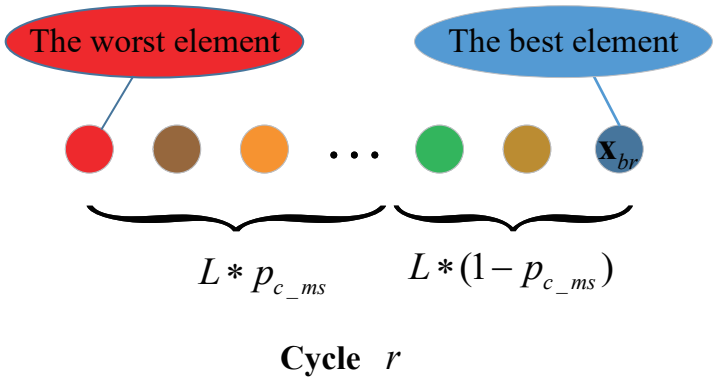


Figure 8. Ordering of elements in a cycle.

Additionally, we use parameter p_{m_ms} to determine the mutation probability during the mid-stage, which is applied in local search operations on the best elements within the cycle, i.e., $\mathbf{x}_{br}(g)$. This targeted search refines this top element further, optimizing its performance characteristics and enhancing its contribution to the overall solution quality.

The selection criteria of elements during this stage become more stringent, focusing on elements that not only exhibit high performance but also contribute to maintaining diversity within the population. This ensures that while the search is becoming more focused, it does not converge prematurely or narrow down excessively, which could potentially exclude optimal solutions.

3.7.3. Late Stage: Intensifying Exploitation

In the late stage of the algorithm, the focus should shift to exploitation, which involves further optimizing the current better solutions $\mathbf{x}_{cbs}(g)$ to accelerate convergence. At this stage, the current better solutions could be the current non-dominated solution $\mathbf{x}_{nd}(g)$ or the best solution within each cycle $\mathbf{x}_{br}(g)$. Furthermore, in the late stage of evolution, the current population $\mathbf{x}_{lr}(g)$ may have already evolved into relatively good solutions. In this study, we determine $\mathbf{x}_{cbs}(g)$ through experimental results and then employ the mutation operator on them for exploitation.

The pseudocode in Algorithm 2 illustrates how these stages are implemented in the algorithm.

Algorithm 2 Stage-Specific strategies at different evolutionary stages

if $c < c_{max} * 20\%$ **then**
  $\mathbf{x}_{new}(g) \leftarrow \texttt{crossover}(\mathbf{x}_{lr}(g), \mathbf{x}_{br'}(g))$
  if $rand < p_{m\_is}$ **then**
    $\mathbf{x}_{new}(g) \leftarrow \texttt{mutation}(\mathbf{x}_{new}(g))$
  end if
else if $c < c_{max} * 80\%$ **then**
  if $l < L * p_{c\_ms}$ **then**
    $\mathbf{x}_{new1}(g) \leftarrow \texttt{crossover}(\mathbf{x}_{lr}(g), \mathbf{x}_{nd}(g))$
  else
    $\mathbf{x}_{new1}(g) \leftarrow \texttt{crossover}(\mathbf{x}_{lr}(g), \mathbf{x}_{br'}(g))$
  end if
  if $rand < p_{m\_ms}$ **then**
    $\mathbf{x}_{new2}(g) \leftarrow \texttt{mutation}(\mathbf{x}_{br}(g))$
  end if
else
  $\mathbf{x}_{new}(g) \leftarrow \texttt{mutation}(\mathbf{x}_{cbs}(g))$
end if
Output the final optimization results

3.7.4. Flowchart of Stage-Specific Strategies of Evolution

The flowchart of Stage-Specific strategies of evolution is shown in Figure 9.

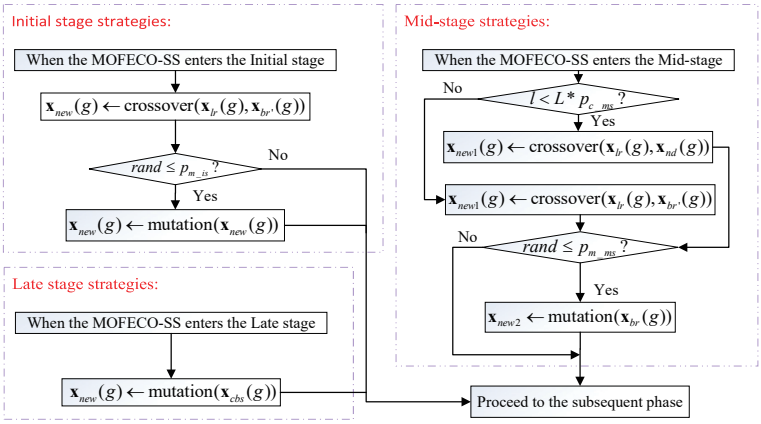


Figure 9. The flowchart of Stage-Specific strategies.

3.8. Flowchart of MOFECO-SS

The process for solving BIGVRP using the MOFECO-SS is depicted in the flowchart as Figure 10. The process unfolds as follows:

- Parameter setting: Initialize by setting all necessary parameters required for the algorithm to function effectively.
- Population initialization: Begin by generating an initial population randomly. This population is of size *popsize* ,which serves as the basis for further evolutionary operations.
- Objective function evaluation: Calculate the objective function for each member of the initial population. Assess the quality of each element based on these calculations.
- Non-dominated sorting: Employ non-dominated sorting on the initial population based on the results of the objective function evaluation. This step identifies a set of non-dominated solutions, which will be utilized in subsequent evolutionary processes.
- Main evolutionary loop:
  - Force calculation: Compute the force exerted on each element within the population. This metric will guide the sorting and selection process.
  - Element sorting: Sort elements within each cycle based on the magnitude of the calculated forces, aligning the population for targeted evolutionary strategies.
  - Evolutionary strategy application: Implement various evolutionary strategies tailored to the elements, depending on their current stage within the evolutionary cycle.
  - Combination of populations: Merge the newly derived individuals from these evolutionary strategies with the existing parent population to form a comprehensive pool of candidates.
- Population management:
  - Further non-dominated sorting: Apply non-dominated sorting to the combined population and compute the crowding distance to ensure a diverse set of solutions.
  - From this sorted and crowded population, extract the top *popsize* elements. These selected elements will constitute the new, updated population for the next iteration of the loop.
- Termination check: Continue the iterative process until the termination condition is satisfied, which in this case is defined as reaching the maximum number of objective function evaluations, c_{max}.
- Output the solution: At the conclusion of the algorithm, once the termination condition is met, output the final set of non-dominated solutions. This set represents the optimal solutions derived from the algorithm.

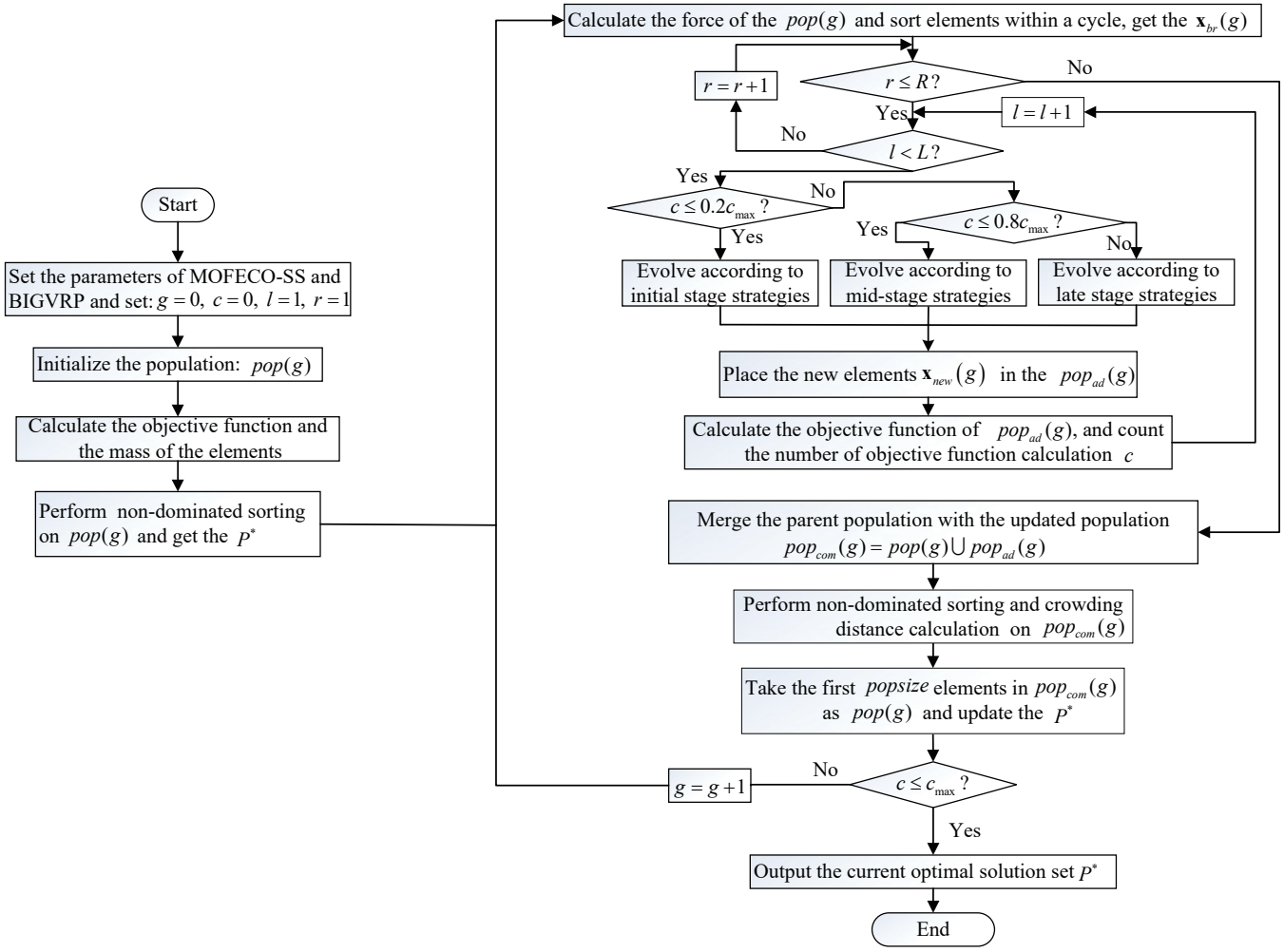


Figure 10. The flowchart of MOFECO-SS.

This structured approach allows the MOFECO-SS to effectively navigate the complexities of optimizing logistics in cold chain management, ensuring that each step is geared towards refining the solution space for the best possible outcomes.

4. Experimental Result and Analysis of MOFECO-SS

To ensure the robustness and reliability of the parameter settings, all experiments for parameter tuning were independently conducted 15 times. For the comparison of algorithms, each experiment was repeated 31 times. The implementation was carried out using MATLAB R2018a and executed on a system with a 2.4 GHz Intel Xeon-E5645 processor, 32 GB of RAM, and running Windows 10.

4.1. Case Study and Parameter Setting

This study applies the proposed model to a real-world scenario referenced from Shandong Jiajiayue Group Co., Ltd., as described in Reference [30]. Jiajiayue is a leading supermarket chain in Shandong Province, recognized for its extensive logistics operations and commitment to integrating agricultural production with retail. For this case, we focus on the Songcun Fresh Logistics Distribution Center in Wendeng, Weihai, China. This center serves as a critical platform for the company's "supermarket + base" model, ensuring efficient and timely delivery of fresh produce.

The case study involves the distribution of vegetables such as cucumbers, tomatoes, celery, and beans to 20 supermarket stores within a 40-km radius of the Wendeng urban area. Deliveries are carried out using a uniform fleet of refrigerated trucks, with a set

average speed of 25 km/h per truck. The external temperature during transportation is assumed to be constant at 27 °C, and the refrigeration temperature is maintained at 6 °C to preserve the quality of the produce.

The location, demand, optimal time window, acceptable time window, and service time for each customer point are known, as shown in Table 2. Number 0 represents the distribution center, and numbers 1–20 represent the 20 customer points. The parameter settings for the BIGVRP are presented in Table 3, while the parameter configurations of MOFECO-SS are detailed in Table 4.

Table 2. Demand information of customers.

Number	*X* Coordinate (km)	*Y* Coordinate (km)	Demand (t)	$[ET, LT]$	$[EET, LLT]$	Service Time (min)
0	13,271.60	2896.72	0.00	[5 : 30, 17 : 00]	[5 : 00, 17 : 30]	0
1	13,270.70	2898.86	1.50	[6 : 00, 8 : 00]	[5 : 30, 9 : 00]	20
2	13,270.47	2900.73	0.50	[7 : 30, 9 : 00]	[7 : 00, 9 : 30]	10
3	13,269.09	2899.42	1.50	[6 : 00, 8 : 00]	[5 : 30, 8 : 30]	20
4	13,268.75	2898.41	2.00	[6 : 30, 8 : 20]	[6 : 00, 9 : 00]	20
5	13,271.67	2901.61	2.00	[6 : 40, 8 : 30]	[6 : 10, 10 : 00]	25
6	13,269.14	2901.44	1.80	[7 : 00, 9 : 00]	[6 : 30, 10 : 20]	25
7	13,267.98	2900.32	1.00	[7 : 20, 9 : 00]	[7 : 00, 9 : 30]	22
8	13,270.21	2902.49	1.00	[7 : 30, 9 : 00]	[7 : 00, 10 : 00]	15
9	13,267.91	2898.22	1.00	[7 : 00, 8 : 30]	[6 : 40, 9 : 30]	15
10	13,266.67	2900.79	1.00	[7 : 00, 9 : 00]	[6 : 30, 9 : 40]	15
11	13,267.42	2902.81	0.50	[7 : 30, 9 : 30]	[7 : 00, 10 : 30]	15
12	13,269.22	2903.54	0.50	[7 : 30, 9 : 00]	[7 : 00, 10 : 00]	10
13	13,265.98	2902.38	1.50	[7 : 30, 9 : 30]	[7 : 00, 10 : 30]	10
14	13,273.00	2901.03	2.00	[7 : 30, 9 : 00]	[7 : 00, 10 : 00]	20
15	13,272.98	2902.44	1.50	[6 : 50, 8 : 30]	[6 : 20, 9 : 30]	25
16	13,271.86	2903.3	2.00	[7 : 00, 8 : 40]	[6 : 40, 9 : 30]	20
17	13,271.00	2902.4	1.50	[7 : 00, 8 : 40]	[6 : 40, 9 : 30]	20
18	13,272.03	2901.11	0.50	[7 : 50, 9 : 00]	[7 : 00, 10 : 00]	10
19	13,269.82	2898.65	2.50	[6 : 30, 8 : 30]	[6 : 00, 9 : 30]	30
20	13,271.21	2898.11	1.00	[7 : 50, 9 : 00]	[7 : 00, 10 : 00]	15

Table 3. Parameters setting of BIGVRP.

Symbols	Unit	Value
N	none	20
K	none	3
c_f	RMB/car	200
c_t	RMB/km	2.0
c_d	RMB/t	1000
c_{r_1}	RMB/h	15
c_{r_2}	RMB/h	20
c_{p_1}	RMB/h	50
c_{p_2}	RMB/h	80
c_e	RMB/kg	1
Q	t	9
Q_c	kg	25
ρ^*	kg/km	0.377
ρ_0	kg/km	0.165
ε_1	none	0.002
ε_2	none	0.003
v	km/h	25
μ	g/kg · h	0.165
η	kg/L	2.63

Table 4. Parameters setting of MOFECO-SS.

Symbols	Value
$popsize$	100
c_{max}	100,000
p_{m_is}	0.9

4.2. Performance Metrics

Multi-objective optimization (MOO) algorithms are employed to address issues involving several conflicting objectives. The primary aim is to generate a range of solutions that achieve an optimal balance between these competing goals. Assessing the performance of MOO algorithms is essential to determining their capability in discovering high-quality solutions that span the entire Pareto front, encompassing all non-dominated solutions.

This study adopts several evaluation metrics to assess the performance of the MOO algorithms:

4.2.1. Hypervolume

The hypervolume (HV) metric is commonly employed to assess the coverage of the objective space by a set of solutions. It provides an integrated view of both the convergence and the diversity of the solution set, offering a balanced perspective on algorithm performance [31]. The HV is calculated using the following formula:

$$HV = \lambda \left(\bigcup_{i=1}^{|s|} v_i \right) \tag{29}$$

where the followins true:

- λ represents the Lebesgue measure.
- v_i is the hypervolume contribution of the solution point, calculated from the reference point.
- s stands for the solution set representing the Pareto front.

The reference point (Ref) is typically chosen based on the worst values for each objective function across the true Pareto front. The HV increases as the solution set provides better coverage of the objective space, indicating improved algorithm performance.

4.2.2. Inverse Generational Distance

The Inverted Generational Distance (IGD) is a widely used metric in multi-objective optimization for assessing algorithmic performance [32]. It measures the average distance from each point on the Pareto front to its nearest solution. The formula for calculating the IGD is as follows:

$$IGD(A, B) = \frac{1}{|B|} \sum_{\mathbf{y} \in B} \min_{\mathbf{x} \in A} d(\mathbf{x}, \mathbf{y}) \tag{30}$$

where:

- A is the set of solutions obtained from the optimization algorithm.
- B is the reference set, it can be the true or an approximate Pareto front.
- $d(\mathbf{x}, \mathbf{y})$ is the distance between a solution $\mathbf{x}$ in set A and a solution $\mathbf{y}$ in set B. This distance is usually calculated using the Euclidean distance.

This metric assesses how well the solutions in set A approximate those in set B, with lower values indicating better approximation. It can assess whether the solution set uniformly and tightly covers the Pareto front.

4.2.3. I_ϵ indicator

The I_ϵ indicator primarily measures the extent to which one solution set can ϵ-dominate another solution set [33]. This means for every solution in the reference set,

there is at least one solution in the algorithm's set such that it is not worse than the reference solution by more than a fixed factor ϵ across all objectives. The formula for the I_ϵ is given as:

$$I_\epsilon(A, B) = \inf\{\epsilon \geq 0 : \ \forall \mathbf{y} \in B, \exists \mathbf{x} \in A : \ \mathbf{x} \leq \epsilon \cdot \mathbf{y}\} \tag{31}$$

where $\mathbf{x} \leq \epsilon \cdot \mathbf{y}$ indicates that for all objectives i, $x_i \leq \epsilon \cdot y_i$.

This definition seeks the smallest ϵ such that for every solution $\mathbf{y}$ in the reference set B. There exists at least one solution $\mathbf{y}$ in set A, and when each objective of $\mathbf{x}$ is scaled by ϵ, it does not perform worse than $\mathbf{y}$.

Therefore, we can use this metric to compare the performance of two solution sets by calculating $I_\epsilon(A, B)$ and $I_\epsilon(B, A)$: If $I_\epsilon(A, B) < I_\epsilon(B, A)$, then A is generally considered better than B because it requires smaller improvements to match or surpass all objectives in B. If $I_\epsilon(A, B) > I_\epsilon(B, A)$, then B is generally considered better than A.

In this paper, we adopt this metric to evaluate the performance differences of algorithms under different parameter settings, thereby determining the values of parameters.

4.2.4. Generational Distance

Generational Distance (GD) is a performance metric used in the field of multi-objective optimization to quantify how close the solutions generated by an optimization algorithm are to the true or reference Pareto front [34]. This metric focuses on the convergence aspect of the solutions, providing a measure of the average minimum distance from each solution in the algorithm-generated set to the nearest point on the reference Pareto front. The formula is given as flow:

$$GD(A) = \frac{1}{|A|} \sqrt{\sum_{i=1}^{|A|} (\min_{\mathbf{y} \in B} d(\mathbf{x}_i, \mathbf{y})^2} \tag{32}$$

This metric evaluates the proximity of the generated solutions to the Pareto front, where smaller values suggest a closer alignment.

4.2.5. Pure Diversity

In the context of multi-objective optimization, the Pure Diversity (PD) metric is used to assess the level of diversity within the solution set generated by the algorithm, focusing on the variations between the different solutions [31]. This metric plays a critical role in evaluating the algorithm's effectiveness in exploring the entire solution space. The formula commonly used to compute PD is as follows:

$$PD = \frac{2}{|A|(|A| - 1)} \sum_{i=1}^{|A|-1} \sum_{j=i+1}^{|A|} d(\mathbf{x}_i, \mathbf{x}_j) \tag{33}$$

where the following is true:

- $\mathbf{x}_i, \mathbf{x}_j$ represent two distinct solutions from the set.
- $d(\mathbf{x}_i, \mathbf{x}_j)$ is the distance between these two solutions, which can be determined using Euclidean distance or any other suitable metric.

This formula provides the average distance between all pairs of solutions within the set, offering a quantitative measure of the diversity in the solution set. A higher PD value indicates greater diversity, reflecting more pronounced differences between the solutions.

4.2.6. Reference Set

From Sections 4.2.2 and 4.2.4, we can see when calculating performance metrics IGD and GD, a reference set is required, which can either be the true Pareto front or an approximate Pareto front. For practical problems in this paper, the multi-objective cold chain logistics distribution and the true Pareto front is unknown. Therefore, to compute IGD and GD, it is necessary to design a reference set to quantitatively evaluate

the performance of algorithms. In this paper, according to the BIGVRP model described, our designed reference set is shown in Figure 11. It is a segment of the curve from the function in Equation (34), where the vertical coordinates range between 0 and 1. From this curve segment, 1000 evenly distributed reference points are selected.

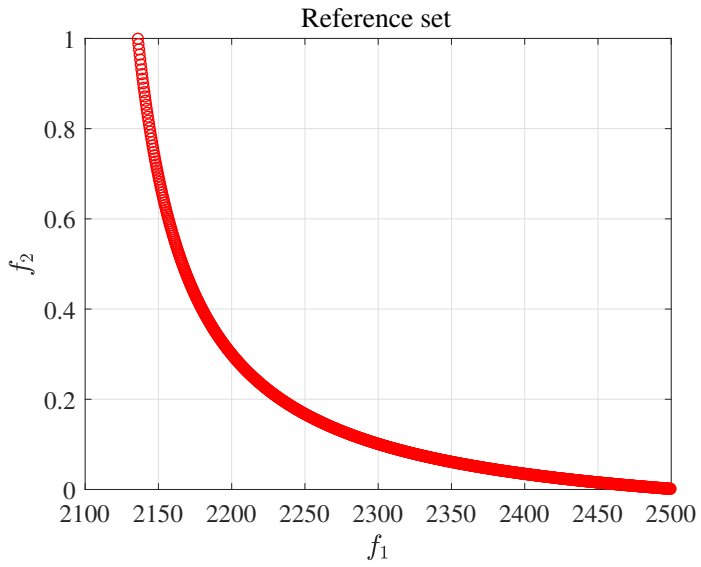


Figure 11. Reference set.

$$f_2 = \frac{40}{f_1 - 2100} - 0.1 + \varepsilon \tag{34}$$

where ε is a small positive number to ensure that Equation (30) remains meaningful; in this study, the value is set to 0.000001.

4.3. Strategy Selection in Late Stage

In the late stage of the MOFECO-SS, a more detailed search around the currently found best solutions, which could be $\mathbf{x}_{nd}(g)$, $\mathbf{x}_{lr}(g)$, or $\mathbf{x}_{br}(g)$, is conducted through mutation operators to find better solutions within high-quality regions of the solution space.

We designed three experiments, where MOFECO-SS employs three different evolutionary strategies during the final phase of evolution to solve the BIGVRP as shown in Table 5. These are used to determine which strategy ultimately enhances the algorithm's performance the most. The solution sets obtained are compared using I_ϵ metric. The calculation results are shown in Table 6:

Table 5. Strategy selection in late stage.

	Evolutionary Strategy
S_1	$\mathtt{Mut}(\mathbf{x}_{nd}(g))$
S_2	$\mathtt{Mut}(\mathbf{x}_{lr}(g))$
S_3	$\mathtt{Mut}(\mathbf{x}_{br}(g))$

Table 6. The average value of indicator $I_\epsilon(A, B)$.

B \ **A**	S_1	S_2	S_3
S_1	\	1.28	1.00
S_2	1.03	\	1.02
S_3	1.94	2.48	\

From Table 6, it can be seen that:

$I_\epsilon(S_1, S_2) = 1.03$ and $I_\epsilon(S_2, S_1) = 1.28$: S_1 needs a 3% improvement to match S_2, while S_2 only requires at least 28% to match S_1. This suggests that S_1 is generally stronger than S_2.

$I_\epsilon(S_1, S_3) = 1.94$ and $I_\epsilon(S_3, S_1) = 1.00$: S_1 needs a 94% improvement to match S_1, while S_3 only requires a 0% to 1% improvement to match S_1. This strongly indicates that S_3 is better than S_1 in achieving the objectives.

$I_\epsilon(S_2, S_3) = 2.48$ and $I_\epsilon(S_3, S_2) = 1.02$: S_2 requires a 148% improvement to reach the levels of S_3, whereas S_3 needs only a slight improvement of 2% to match S_2. This suggests that S_3 is significantly better than S_2.

In summary, S_3 appears to be the most competitive solution set among the three when compared against the others, as it generally requires less improvement to surpass the other sets or is already performing better. Indeed, in the late stage of evolution, the population may have already evolved into relatively good individuals. Typically, mutating the current non-dominated solutions $\mathbf{x}_{nd}(g)$ or the optimal solutions within a cycle $\mathbf{x}_{br}(g)$ might be more promising as they represent potential global optima or local optima. Mutating all individuals $\mathbf{x}_{lr}(g)$ could lead to excessive exploration of the search space, reducing the efficiency of the algorithm. If local search is applied to the set of non-dominated solutions at this stage, it could likely lead to local optima, reducing the diversity of the algorithm. However, conducting local searches on the optimal individuals within a cycle during the late stage of evolution will not only avoid the algorithm getting trapped in local optima but also enhance its diversity and make it easier to find globally optimal solutions.

Therefore, in the late stage of MOFECO-SS evolution, we employ local search on $\mathbf{x}_{br}(g)$ to enhance exploitation.

4.4. Parameters Study

- Determining the parameters L and R
  In the MOFECO-SS algorithm, the population is divided into R cycles, each consisting of L elements, thus $popsize = L \times R$. In this paper, $popsize = 100$, it is necessary to study the impact of the number of elements in each cycle on the performance of the algorithm and to determine the optimal number of elements. Therefore, we established five sets of experiments based on different values of L, with L increasing from the smallest to the largest. Each set of experiments was conducted independently 15 times. After running these experiments, we computed the average values of the indicator I_ϵ from the 15 outcomes. The averages are presented in Table 7.
  As a consequence, the algorithm for BIGVRP performs best when $L = 5$, $R = 20$. This indicates the specific configuration of the parameters, where the number of elements per cycle is set to 5 and the number of cycles is set to 20. The results in the optimal performance of the MOFECO-SS for solving the BIGVRP. This finding is significant because it suggests that both a smaller grouping of elements within each cycle and a higher total number of such cycles lead to more effective outcomes.
  The optimal results from this specific configuration may be due to the interaction between L and R, which influences the algorithm's capacity to effectively explore and exploit the solution space. For instance, with $L = 5$, each cycle contains a manageable number of elements, potentially allowing the algorithm to more effectively fine-tune solutions within each cycle. Additionally, having $R = 20$ cycles increases the diversity of the solutions being explored, which may help escape local optima and speed up the overall search process.
- Determining the parameters p_{c_ms} and p_{m_ms}
  During the mid-stage of evolution in MOFECO-SS, we used two parameters, p_{c_ms} and p_{m_ms}, to balance exploration and exploitation. The parameter p_{c_ms} divides the elements within each cycle into two parts, each adopting different strategies to explore new solutions. The parameter p_{m_ms}, on the other hand, determines the probability of exploiting already discovered good solutions. Therefore, we set different values for these two parameters and conducted experiments, still using the indicator I_ϵ to

compare the performance of the algorithm based on different settings. The calculation results are shown in Tables 8 and 9, respectively.

Table 7. The average value of indicator $I_\epsilon(A, B)$.

B \ A	$L = 5$	$L = 10$	$L = 20$	$L = 25$	$L = 50$
$L = 5$	\	2.11	1.85	1.97	1.82
$L = 10$	1.13	\	1.09	1.12	1.06
$L = 20$	1.24	2.11	\	1.32	1.36
$L = 25$	1.34	1.80	1.40	\	1.56
$L = 50$	1.22	1.36	1.30	1.27	\

Table 8. The average value of indicator $I_\epsilon(A, B)$.

B \ A	$p_{c\_ms} = 0.2$	$p_{c\_ms} = 0.4$	$p_{c\_ms} = 0.6$	$p_{c\_ms} = 0.8$
$p_{c\_ms} = 0.2$	\	1.22	1.17	1.33
$p_{c\_ms} = 0.4$	1.73	\	1.63	1.95
$p_{c\_ms} = 0.6$	1.64	1.65	\	1.90
$p_{c\_ms} = 0.8$	1.29	1.38	1.31	\

Table 9. The average value of indicator $I_\epsilon(A, B)$.

B \ A	$p_{m\_ms} = 0.1$	$p_{m\_ms} = 0.5$	$p_{m\_ms} = 0.9$
$p_{m\_ms} = 0.1$	\	1.08	1.16
$p_{m\_ms} = 0.5$	1.99	\	1.98
$p_{m\_ms} = 0.9$	1.64	1.41	\

From Table 7, the following can be observed:

- $I_\epsilon(L = 5, L = 10) < I_\epsilon(L = 10, L = 5)$,
- $I_\epsilon(L = 5, L = 20) < I_\epsilon(L = 20, L = 5)$,
- $I_\epsilon(L = 5, L = 25) < I_\epsilon(L = 25, L = 5)$,
- $I_\epsilon(L = 5, L = 50) < I_\epsilon(L = 50, L = 5)$.

From Table 8, it can be observed that when A represents the solution set obtained with $p_{c\_ms} = 0.6$ and B represents the solution sets obtained under all other conditions, $I_\epsilon(A, B)$ is consistently smaller than $I_\epsilon(B, A)$. This indicates that the solutions derived from setting $p_{c\_ms} = 0.6$ are superior, suggesting that this parameter setting may be more effective or efficient compared with others.

This indicates that during the mid-stage of MOFECO-SS, achieving the best results involves crossing the top 60% of elements in each cycle with $\mathbf{x}_{nd}$, while the remaining 40% of elements undergo crossover with the best elements from other cycles outside their own.

Table 9 compares the performance of different mutation probabilities against each other using the indicator $I_\epsilon(A, B)$. It can be observed that when A represents the solution set obtained with $p_{m\_ms} = 0.5$ and B represents the solution sets obtained under all other conditions, $I_\epsilon(A, B)$ is consistently smaller than $I_\epsilon(B, A)$, which suggests that a mutation probability of $p_{m\_ms} = 0.5$ offers the most balanced and effective approach during the mid-stage of an evolutionary algorithm. This rate effectively balances exploration (finding new solutions) and exploitation (refining existing solutions), leading to better overall performance compared with both higher and lower mutation rates.

Choosing this mutation rate helps ensure that the algorithm neither converges too quickly to local optima (a risk with lower mutation rates) nor explores too inefficiently (a risk with higher mutation rates). Such balanced settings are crucial for achieving robust and efficient performance in MOFECO-SS.

4.5. Comparison with Optimization Algorithms

To further validate the performance of the MOFECO-SS algorithm proposed in this paper, we also employed seven other multi-objective optimization algorithms to solve the BIGVRP under the same conditions. Among these, four are multi-objective optimization algorithms introduced in recent years based on different mechanisms: Multi-Objective Grey Wolf Optimization (MOGWO) [35], Multi-Objective Whale Optimization Algorithm (MOWOA) [36], Multi-Objective Honey Badger Algorithm (MOHBA) [37], and Multi-Objective Aquila Optimization (MOAO) [38]. In addition, there is the classic NSGA-II, and the last two are the original MOFECO and an improved version based on local search, LSMOFECO. The selection of these algorithms aims to cover a range of classic to modern algorithm strategies and mechanisms, providing a comprehensive assessment of the performance and practicality of the MOFECO-SS algorithm in the field of multi-objective optimization.

The seven multi-objective optimization algorithms are applied to solve the BIGVRP. Each algorithm maintains the same population size and evolutionary termination conditions as MOFECO-SS, with other parameter settings taken from their original proposals. Each algorithm is independently run 31 times, and performance is analyzed using calculated evaluation metrics.

Table 10 shows that MOFECO-SS excels in HV, IGD, and PD metrics and is second only to MOFECO in GD. This suggests that MOFECO-SS's solution sets, devised to tackle the BIGVRP, demonstrate robust convergence and diversity. Meanwhile, algorithms like MOAO, MOHBA, MOWOA, and MOGWO generally perform poorly across all metrics, indicating they may be less suited for the BIGVRP without further adaptation. Table 11 confirms MOFECO-SS's superiority in HV and IGD, while MOFECO leads in GD and PD. Table 12 reveals the standard deviation of these metrics over 31 runs, noting MOHBA's unusually low standard deviation, likely due to fewer solution findings. Despite this, MOFECO-SS consistently outperforms in HV, IGD, and GD, with LSMOFECO excelling in PD.

Table 10. Comparison of eight algorithms based on the average values of various performance metrics.

	HV	***IGD***	***GD***	***PD***
NSGAII	342.87	77.85	0.09	8642.52
MOAO	138.09	166.58	14.66	5781.14
MOHBA	102.51	197.96	31.10	4969.43
MOWOA	157.71	150.47	8.93	7218.92
MOGWO	186.29	135.46	5.79	7031.24
MOFECO	344.81	69.88	0.05	11,309.38
LSMOFECO	329.24	73.12	0.14	10,153.59
MOFECO-SS	383.65	54.55	0.08	13,771.46

In addition, Figure 12 presents the Pareto solution sets obtained by different algorithms, demonstrating the effectiveness of the MOFECO-SS algorithm in finding superior solutions. The superiority of MOFECO-SS is evident in several aspects:

- Distribution and convergence:
  The Pareto front solutions generated by MOFECO-SS exhibit better distribution and convergence compared with other algorithms. The solutions are more uniformly spread across the Pareto front, indicating a broader range of high-quality solutions.
- Objective function values:
  The axes on the graph represent the two objective functions. The solutions obtained by MOFECO-SS are closer to the optimal values, as indicated by the lower total costs and higher customer satisfaction. This demonstrates that the solutions found by MOFECO-SS are more efficient and effective in balancing the trade-offs between the two objectives.

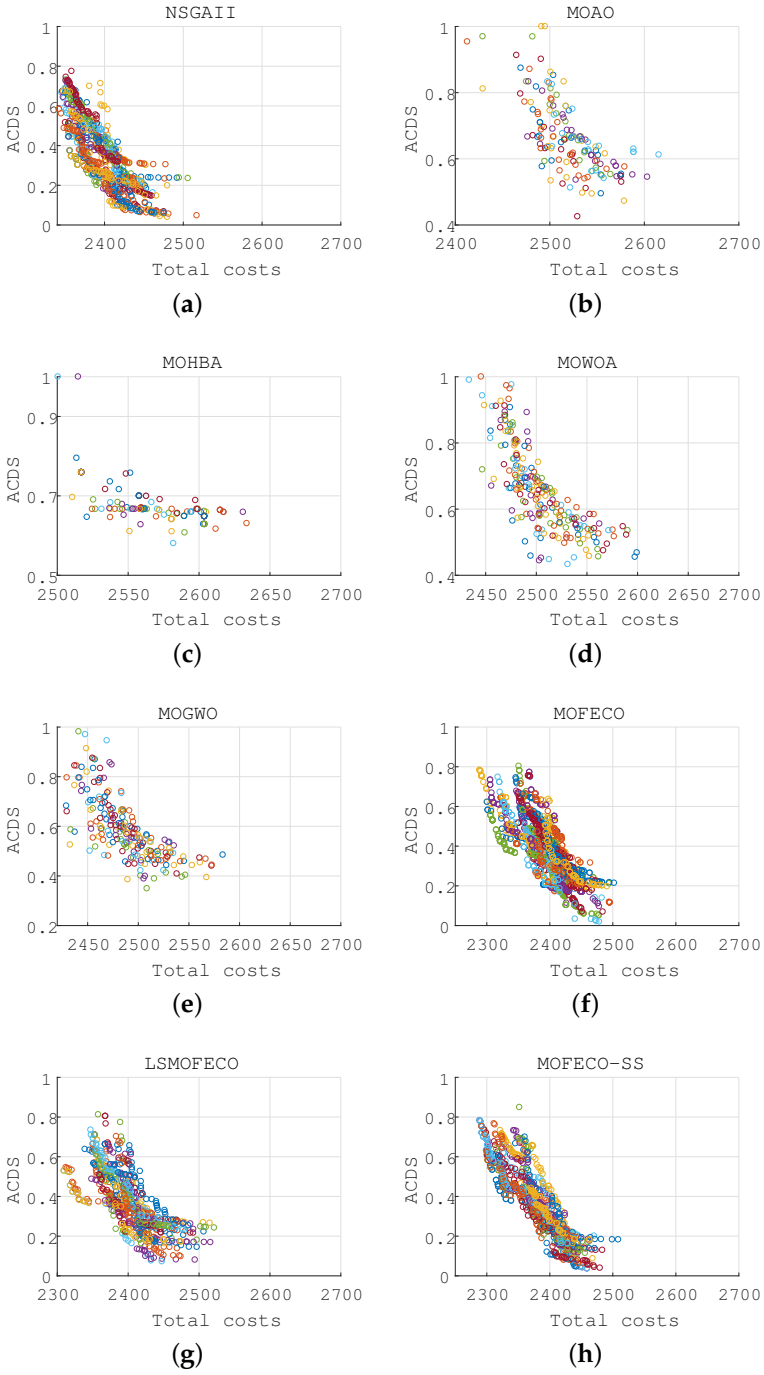


Figure 12. Pareto fronts obtained by eight algorithms in 31 independent runs (solutions obtained in each run are denoted by each color). (**a**) NSGAII, (**b**) MOAO, (**c**) MOHBA, (**d**) MOWOA, (**e**) MOGWO, (**f**) MOFECO, (**g**) LSMOFECO, (**h**) MOFECO-SS.

In summary, Figure 12 illustrates that the proposed MOFECO-SS algorithm outperforms the other seven algorithms by providing better-distributed and more convergent Pareto solutions with improved objective function values. Although some algorithms like NSGA-II, MOFECO, and LSMOFECO show decent distribution and convergence, the objective function values found by these algorithms are generally worse compared with those found by MOFECO-SS. Overall, MOFECO-SS demonstrates superior performance in both the distribution and convergence of solutions, validating its capability to find higher-quality solutions in the context of the BIGVRP in CCL.

Table 11. Comparison of eight algorithms based on the best values of various performance metrics.

	HV	*IGD*	*GD*	*PD*
NSGAII	400.02	63.68	0.04	18,605.93
MOAO	173.60	109.35	1.18	11,700.03
MOHBA	114.39	182.29	20.26	7710.19
MOWOA	175.63	123.02	0.33	15,234.82
MOGWO	206.89	119.14	0.24	11,680.04
MOFECO	416.41	38.21	0.03	20,834.71
LSMOFECO	421.45	45.73	0.05	14,001.52
MOFECO-SS	427.44	35.93	0.04	20,664.21

Table 12. Comparison of eight algorithms based on the standard deviation of various performance metrics.

	HV	*IGD*	*GD*	*PD*
NSGAII	31.68	11.17	0.08	3174.46
MOAO	11.96	19.76	7.40	2585.58
MOHBA	4.29	3.19	4.91	1087.23
MOWOA	9.70	11.48	5.39	2742.83
MOGWO	11.85	10.03	4.67	2254.26
MOFECO	33.30	16.77	0.02	4282.28
LSMOFECO	35.98	14.79	0.18	2975.77
MOFECO-SS	24.62	13.93	0.04	3461.99

Based on this, Figures 13–16 display the median value of the performance metrics from 31 experimental results for each algorithm, providing a clearer comparison of these eight algorithms. Additionally, the reference set designed for calculating the IGD and GD evaluation metrics is also depicted. The four figures visually demonstrate that a majority of the solutions discovered by MOFECO-SS lie in the lower-left region compared with solutions found by other algorithms. This is particularly advantageous for the minimization model proposed in this paper, as the solutions identified by MOFECO-SS are generally superior.

Overall, MOFECO-SS's superior performance across most metrics can be attributed to its effective integration of multi-objective optimization strategies with a stable and well-balanced search algorithm, capable of consistently finding near-optimal solutions with high diversity and dominance in the solution space for the BIGVRP.

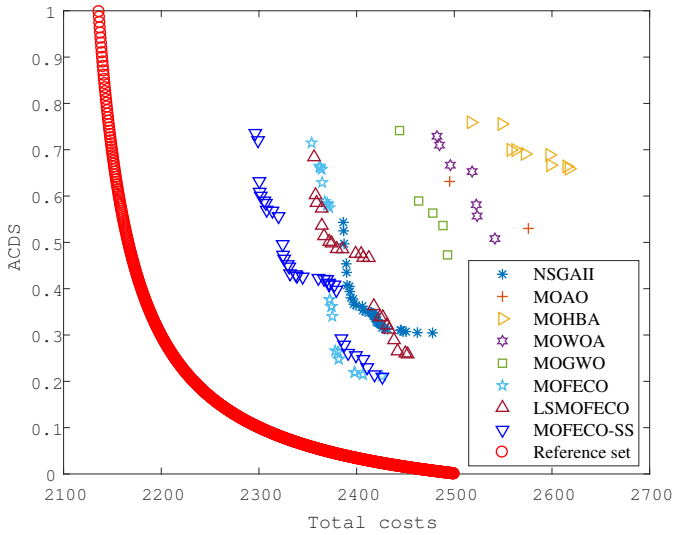


Figure 13. Pareto fronts of each algorithm corresponding to the median HV value from 31 experimental results.

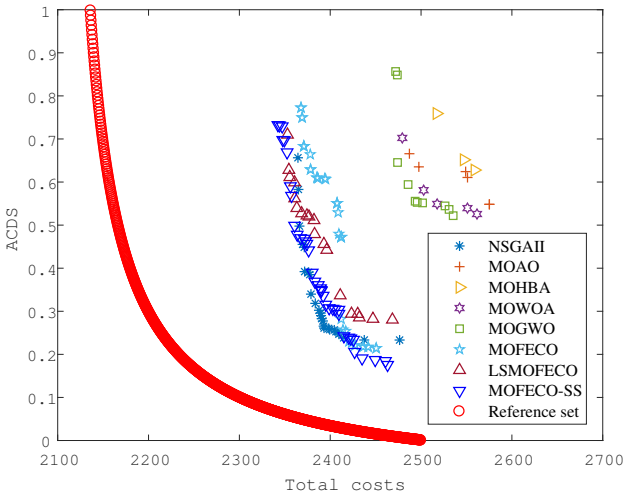


Figure 14. Pareto fronts of each algorithm corresponding to the median *IGD* value from 31 experimental results.

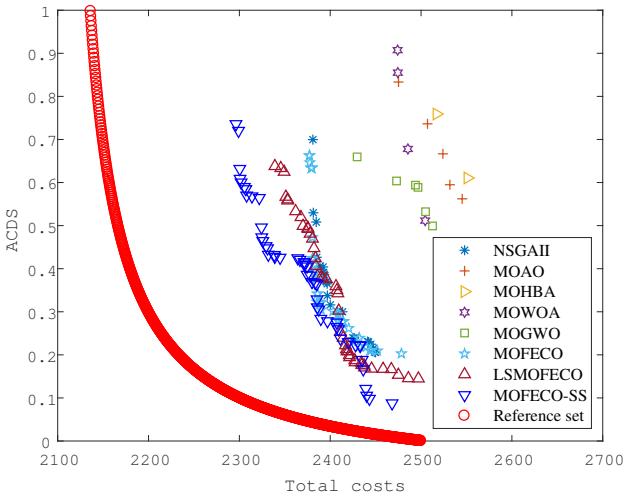


Figure 15. Pareto fronts of each algorithm corresponding to the median *GD* value from 31 experimental results.

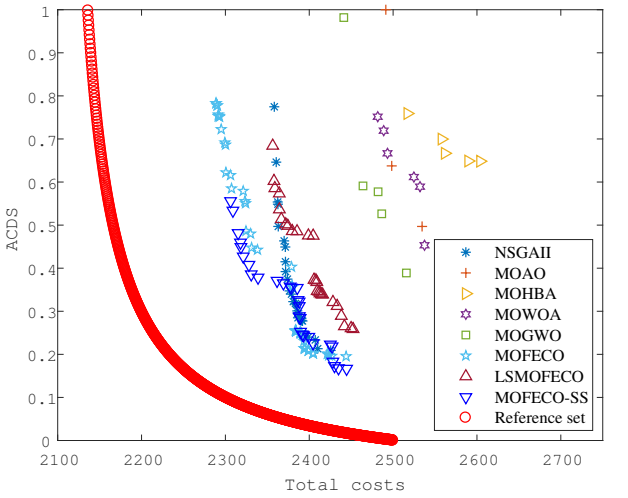


Figure 16. Pareto fronts of each algorithm corresponding to the median *PD* value from 31 experimental results.

4.6. Statistical Significance Testing

To ensure the statistical significance of the performance differences between MOFECO-SS and the other seven algorithms across the four key metrics (HV, IGD, GD, and PD), we performed the Wilcoxon signed-rank test. The results in Table 10 display the average values for each algorithm across the metrics, and Table 13 presents the *p*-values from the Wilcoxon test, allowing us to analyze whether these differences are statistically significant.

In terms of the HV, MOFECO-SS demonstrated superior performance, achieving the highest HV value (383.65) among all the algorithms. The Wilcoxon test results confirmed that the differences between MOFECO-SS and the other algorithms in the HV metric are statistically significant, with a *p*-value far below 0.05. This indicates that MOFECO-SS significantly outperformed the other algorithms in terms of coverage and diversity of solutions.

For the IGD, MOFECO-SS also achieved the best results, with an IGD value of 54.55, which was the lowest among all the algorithms. A lower IGD indicates that the solutions generated by MOFECO-SS are closer to the true Pareto front. The Wilcoxon test further verified that the differences between MOFECO-SS and the other algorithms in IGD are statistically significant, with *p*-values all below 0.05. This confirms that MOFECO-SS produced solutions that are significantly closer to the true Pareto front compared with the other algorithms.

However, in terms of GD, MOFECO performed slightly better than MOFECO-SS, with a GD value of 0.05 compared with MOFECO-SS's 0.08. Although the difference between these two values is relatively small, the Wilcoxon test indicates that the difference is statistically significant ($p = 4.86 \times 10^{-4}$), suggesting that MOFECO has a slight edge over MOFECO-SS in generating solutions closer to the ideal Pareto front.

Finally, for PD, MOFECO-SS demonstrated outstanding performance, with a PD value of 13,771.46, which was significantly higher than that of the other algorithms. The Wilcoxon test confirmed that the differences between MOFECO-SS and the other algorithms in the PD metric are statistically significant, with *p*-values all below 0.05. This indicates that MOFECO-SS generated a more diverse set of solutions compared with the other algorithms.

In summary, the Wilcoxon signed-rank test results demonstrate that MOFECO-SS significantly outperformed the other algorithms in terms of HV, IGD, and PD, indicating superior coverage, proximity to the Pareto front, and solution diversity. While MOFECO was slightly better than MOFECO-SS in the GD metric, the difference is statistically significant. These findings confirm the robustness and effectiveness of MOFECO-SS in solving multi-objective optimization problems, particularly in the context of the BIGVRP, where it excels in generating high-quality, diverse solution sets across multiple key performance metrics.

Table 13. Wilcoxon *p*-values for comparisons between MOFECO-SS and other algorithms on key metrics.

	HV	***IGD***	***GD***	***PD***
NSGAII	1.23×10^{-4}	6.57×10^{-6}	9.97×10^{-2}	9.48×10^{-6}
MOAO	1.17×10^{-6}	1.17×10^{-6}	1.17×10^{-6}	1.43×10^{-6}
MOHBA	1.17×10^{-6}	1.17×10^{-6}	1.17×10^{-6}	1.17×10^{-6}
MOWOA	1.17×10^{-6}	1.17×10^{-6}	1.17×10^{-6}	3.41×10^{-6}
MOGWO	1.17×10^{-6}	1.17×10^{-6}	1.17×10^{-6}	1.92×10^{-6}
MOFECO	1.33×10^{-4}	1.07×10^{-3}	4.86×10^{-4}	2.30×10^{-2}
LSMOFECO	9.48×10^{-6}	1.55×10^{-4}	1.83×10^{-3}	3.36×10^{-4}

4.7. Experimental Results and Compromise Solution

4.7.1. Experimental Results

In this study, we have designed an optimal set of parameters for the MOFECO-SS algorithm, enabling it to achieve the best performance in solving the BIGVRP. We selected a solution set that corresponds to the maximum *HV* indicator for analysis. Depending on different optimization considerations of decision-makers, specific solutions are chosen as shown in Table 14.

Table 14. Solutions corresponding to different decisions.

	f_1 (RMB)	f_2	C_f (RMB)	C_t (RMB)	C_d (RMB)	C_r (RMB)	C_p (RMB)	C_e (RMB)	EM (kg)
Solution_1	2299.44	0.63	600.00	317.96	106.41	178.79	1096.28	54.23	79.23
Solution_2	2442.12	0.05	600.00	328.86	120.84	180.54	1211.89	61.23	86.23
Solution_3	2413.32	0.24	600.00	303.27	111.41	175.85	1222.79	53.94	78.94
Solution_4	2302.31	0.62	600.00	320.37	105.99	178.64	1097.31	54.45	79.45
Solution_5	2414.13	0.22	600.00	304.37	112.93	175.64	1221.20	54.45	79.45

- Solution_1 corresponds to the path with the minimum total and time penalty costs. This makes it ideal for scenarios where minimizing direct and indirect costs related to distribution inefficiencies is crucial.
- Solution_2 is the best for customer satisfaction. This route is optimized for delivering the highest levels of service quality, potentially considering factors like delivery times and customer feedback.
- Solution_3 is selected for having the minimum transportation costs, carbon emissions, and carbon emission costs. It represents the most environmentally friendly option, suitable for operations aiming to reduce their ecological footprint and comply with environmental regulations.
- Solution_4 corresponds to the route with the minimum costs associated with cargo damage. This route would be particularly valuable in transporting fragile or high-value items, where reducing the incidence of damage is critical.
- Solution_5 is the best route for minimizing refrigeration costs. It is ideal for the transport of perishable goods where refrigeration is a major cost driver.

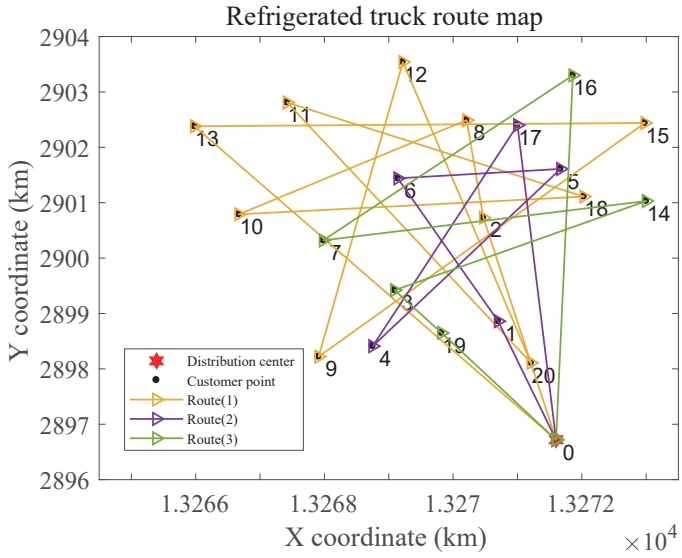
Figures 17–21 illustrate the specific routes for each refrigerated truck corresponding to each logistics path. The choice of route depends on the specific operational goals and constraints of the logistics operation. Each route offers distinct advantages in terms of cost, customer satisfaction, environmental impact, and specific logistical challenges. By aligning route selection with strategic business objectives, whether that is minimizing costs, enhancing customer satisfaction, promoting sustainability, or ensuring product integrity, decision-makers can effectively optimize their logistics operations to support broader business outcomes. This strategic approach ensures that the MOFECO-SS is not only optimized for performance but also tailored to meet the diverse needs of modern logistics environments.

From the experimental results, it can be seen that the findings from this study offer several valuable insights for managers in the cold chain logistics sector:

Balancing cost and sustainability: The MOFECO-SS algorithm provides a practical tool for logistics managers to balance cost efficiency with environmental sustainability. By minimizing carbon emissions alongside distribution costs, managers can align their operations with corporate sustainability goals and regulatory requirements, thereby enhancing the company's reputation and compliance.

Enhancing customer satisfaction: The algorithm's ability to maximize customer satisfaction through optimized delivery routes ensures that perishable goods such as food and pharmaceuticals are delivered timely and in optimal condition. This is critical for maintaining customer loyalty and competitive advantage in markets where product quality and timely delivery are paramount.

Flexibility in route planning: The flexibility of MOFECO-SS allows for tailored route planning that can adapt to various operational constraints and customer demands. Managers can use this flexibility to dynamically adjust routes in response to real-time changes in demand, traffic conditions, and other disruptions, thereby improving the overall resilience and responsiveness of the logistics network.

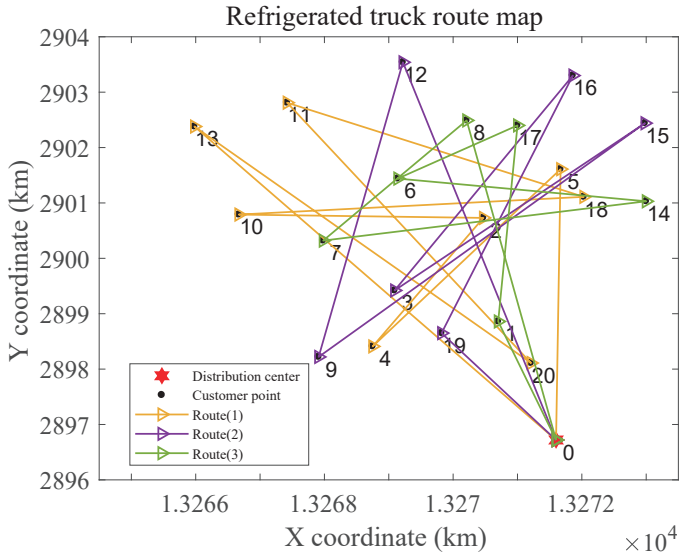


$$\text{Route}(1) = \{0, 13, 15, 9, 12, 20, 11, 18, 10, 8, 2, 0\};$$

$$\text{Route}(2) = \{0, 17, 4, 5, 6, 1, 0\};$$

$$\text{Route}(3) = \{0, 16, 7, 14, 3, 19, 0\}.$$

Figure 17. Solution_1 with the minimum total and time penalty costs.

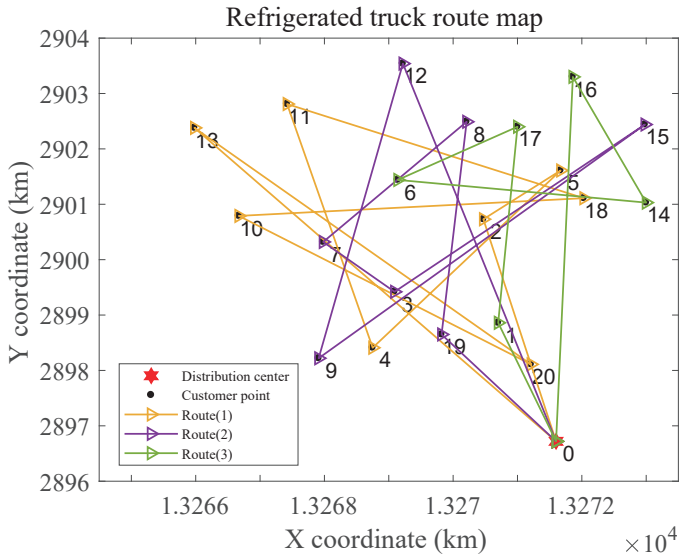


$$\text{Route}(1) = \{0, 13, 20, 11, 18, 10, 2, 4, 5, 0\};$$

$$\text{Route}(2) = \{0, 12, 9, 15, 3, 16, 19, 0\};$$

$$\text{Route}(3) = \{0, 8, 7, 14, 6, 17, 1, 0\}.$$

Figure 18. Solution_2 with the maximum customer satisfaction.

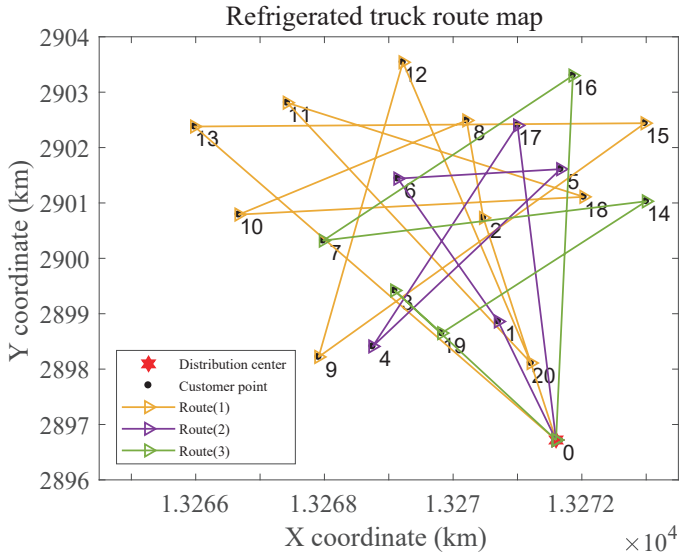


$$\text{Route}(1) = \{0, 13, 20, 10, 18, 11, 4, 5, 2, 0\};$$

$$\text{Route}(2) = \{0, 12, 9, 15, 3, 7, 8, 19, 0\};$$

$$\text{Route}(3) = \{0, 16, 14, 6, 17, 1, 0\}.$$

Figure 19. Solution_3 with the minimum transportation and carbon emission costs.

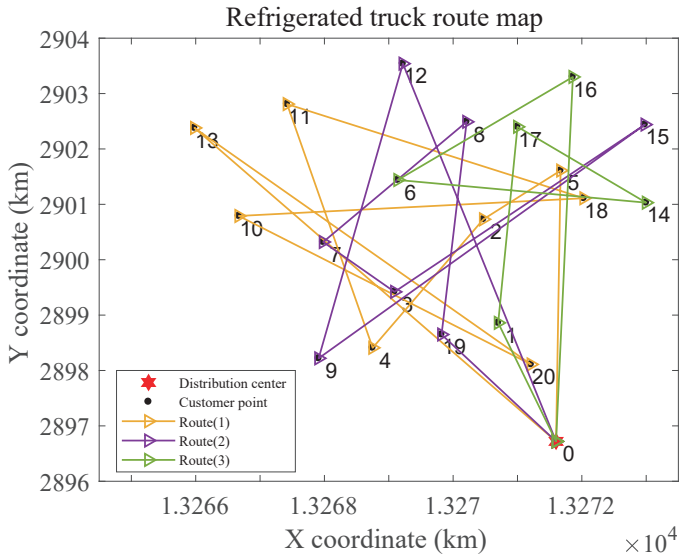


$$\text{Route}(1) = \{0, 13, 15, 9, 12, 20, 11, 18, 10, 8, 2, 0\};$$

$$\text{Route}(2) = \{0, 17, 4, 5, 6, 1, 0\};$$

$$\text{Route}(3) = \{0, 16, 7, 14, 19, 3, 0\}.$$

Figure 20. Solution_4 with the minimum cargo damage costs.



$$\text{Route}(1) = \{0, 13, 20, 10, 18, 11, 4, 2, 5, 0\};$$

$$\text{Route}(2) = \{0, 12, 9, 15, 3, 7, 8, 19, 0\};$$

$$\text{Route}(3) = \{0, 16, 6, 14, 17, 1, 0\}.$$

Figure 21. Solution_5 with the minimum refrigeration costs.

Strategic decision-making: The comprehensive approach of MOFECO-SS in considering multiple objectives simultaneously provides managers with a holistic view of the trade-offs involved in route planning. This facilitates more informed strategic decision-making, enabling managers to prioritize different aspects of logistics operations based on current business objectives and external conditions.

Scalability and adaptability: The algorithm's Stage-Specific adaptation mechanism ensures that it remains effective across different scales of operation, from small local networks to large international logistics systems. Managers can leverage this scalability to implement the algorithm across various levels of their supply chain, ensuring consistent optimization performance.

By integrating these insights into their operations, managers can enhance both the efficiency and sustainability of their logistics networks, ultimately contributing to better operational outcomes and long-term strategic advantages.

4.7.2. Compromise Solution

In multi-objective optimization, different solutions may emphasize different objectives, making it essential to select a compromise solution that balances all key objectives. In this study, we utilize the Ideal Point Method [39] to identify a compromise solution from the set of solutions with the largest HV indicator. The Ideal Point Method allows us to evaluate how close each solution is to the ideal point, where the ideal point represents the best possible values for each of the objectives in the problem.

To begin, we determined the ideal point for each objective function. For objectives that need to be minimized (e.g., transportation costs, carbon emissions), the ideal point is the minimum value observed across all solutions. Conversely, for objectives that need to be maximized, the ideal point is the maximum value observed. Thus, the ideal point serves as a benchmark for optimal performance across all objectives.

Next, we calculated the distance between each solution and the ideal point using the Euclidean distance formula:

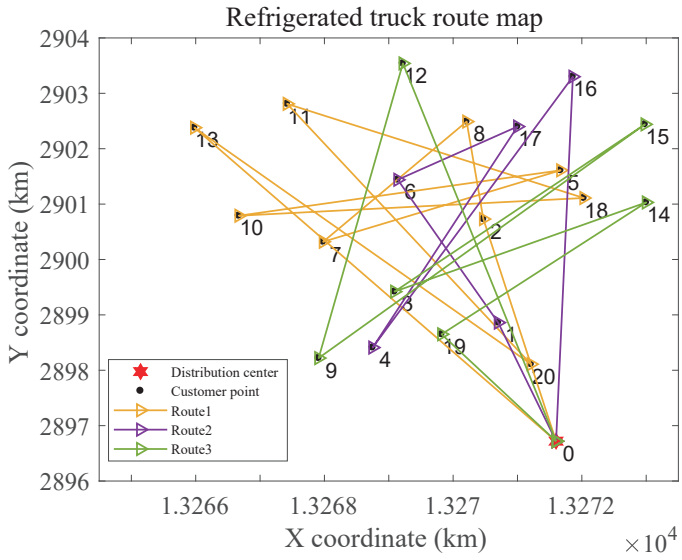
$$d(x) = \sqrt{\sum_{i}^{n} (f_i(x) - f_i^*)^2} \tag{35}$$

where $f_i(x)$ represents the value of the i-th objective for solution x, and f_i^* represents the ideal value for the i-th objective, n is the number of objective functions. By calculating the distance for each solution, we identified the solution closest to the ideal point. This solution, known as the compromise solution, offers the best balance among all objectives.

Table 15 shows the objective function values of the selected compromise solution. This solution balances several objectives, such as total cost, customer satisfaction, and environmental impact. For example:

- The total cost is 2363.55 RMB, which, while not the lowest, represents a well-balanced trade-off with other objectives.
- Customer satisfaction is relatively high, with a score of 0.42, indicating that delivery conditions largely meet customer expectations.
- The carbon emissions are 84.84 kg, demonstrating strong environmental sustainability.

Figure 22 visually illustrates the specific routes for each refrigerated truck corresponding to the compromise solution. The compromise solution aims to balance multiple objectives such as cost, customer satisfaction, and environmental sustainability. Each route is optimized to meet the overall strategic goals of the logistics operation by minimizing trade-offs among these objectives. The three distinct routes demonstrate how different customer points are served efficiently, ensuring that the logistics operation remains both cost-effective and environmentally responsible while maintaining high service quality. This approach allows decision-makers to achieve a well-rounded optimization, addressing various operational constraints and priorities in a balanced manner.



$$\text{Route}(1) = \{0, 13, 20, 11, 18, 10, 5, 7, 8, 2, 0\};$$

$$\text{Route}(2) = \{0, 16, 4, 17, 6, 1, 0\};$$

$$\text{Route}(3) = \{0, 12, 9, 15, 3, 14, 19, 0\}.$$

Figure 22. Compromise solution.

Table 15. Objective function values of the compromise solution.

Objective Function	Values
f_1 (RMB)	2363.55
f_2 (RMB)	0.42
C_f (RMB)	600.00
C_t (RMB)	325.56
C_d (RMB)	114.20
C_r (RMB)	180.31
C_p (RMB)	1143.48
C_e (RMB)	59.84
EM (kg)	84.84

5. Conclusions and Future Work

This study introduces the MOFECO-SS algorithm, designed to balance the dual objectives of minimizing total costs (including carbon emissions) and maximizing customer satisfaction in the context of GVRP. Through extensive experiments, it has been shown that MOFECO-SS significantly outperforms other meta-heuristic algorithms across several key performance metrics. Statistical significance testing via the Wilcoxon signed-rank test confirmed that MOFECO-SS consistently generates higher-quality solutions with better coverage, proximity to the Pareto front, and diversity.

The results from this study indicate that MOFECO-SS produces Pareto front solutions with superior distribution and convergence compared with other algorithms, offering a diverse set of high-quality solutions. By leveraging the flexibility of the MOFECO-SS framework, logistics managers can tailor route plans to align with strategic business objectives, such as reducing operational costs, enhancing customer satisfaction, and promoting sustainability. The application of the Ideal Point Method further demonstrated the algorithm's capability to select compromise solutions from the Pareto front, balancing multiple objectives to meet varying operational demands.

Furthermore, the statistical significance testing confirmed the robustness of MOFECO-SS across all metrics. The algorithm achieved significant improvements in HV, IGD, and PD when compared with other algorithms. Although MOFECO performed slightly better in GD, the overall performance of MOFECO-SS remains competitive and highly effective in addressing multi-objective optimization challenges, particularly in GVRP scenarios.

Despite these promising results, this study acknowledges several limitations. Certain assumptions were made about road conditions, traffic, and customer demand to simplify the problem, which may not fully reflect the complexities of real-world logistics environments. While these assumptions are common in logistics optimization research, future studies should aim to incorporate more dynamic and realistic factors, such as variable traffic conditions and fluctuating customer demand patterns, to enhance the model's applicability.

Additionally, the performance of MOFECO-SS is influenced by the choice of parameters. While this study utilized manually-tuned parameters, future research could explore automated parameter tuning techniques, including machine learning-based methods, to enhance the algorithm's adaptability across different logistics scenarios. This could lead to improved performance and broader applicability, making MOFECO-SS more effective in a wider range of real-world conditions.

Looking ahead, future research could focus on several key areas to further enhance the practicality and applicability of the MOFECO-SS algorithm. First, integrating real-time data, such as traffic information and dynamic customer demands, through Internet of Things (IoT) technologies could significantly improve the algorithm's performance in dynamic logistics environments. By incorporating real-time data, the algorithm could adapt more efficiently to sudden changes in the logistics network, thereby improving route efficiency and customer service levels.

In addition, future studies could explore the application of MOFECO-SS in various industries through case studies and pilot implementations. Testing the algorithm in different sectors, such as perishable goods, pharmaceuticals, and e-commerce, could offer valuable insights into its practical benefits and challenges. These real-world applications would help bridge the gap between theoretical optimization models and practical logistics operations, providing valuable feedback to refine the algorithm for specific industry needs.

Finally, expanding the scope of the optimization objectives by considering additional sustainability factors, such as energy consumption during transportation and the use of alternative fuel vehicles, could further enhance the algorithm's contribution to sustainable logistics operations. By incorporating broader sustainability goals, MOFECO-SS could become an even more powerful tool for addressing both operational and environmental challenges in modern logistics networks.

In conclusion, by addressing these future research directions, this study not only advances the academic understanding of logistics optimization but also provides practical, actionable strategies for logistics managers to improve operational efficiency and sustainability in supply chain operations. The combination of strong performance across key metrics and statistical validation ensures that MOFECO-SS is a valuable contribution to the field of multi-objective optimization.

Author Contributions: Conceptualization, Y.X.; methodology, Y.X. and M.L.; software, Y.X.; formal analysis, Y.X. and J.G.; investigation, Y.X.; data curation, Y.X. and Z.M.; writing—original draft preparation, Y.X.; writing—review and editing, J.G., Z.M. and C.J.; supervision, M.L. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Fundamental Research Funds for the Central Universities, Grant No. 222201917006, under the affiliation of the Key Laboratory of Smart Manufacturing in Energy Chemical Process (East China University of Science and Technology), Ministry of Education.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The original contributions presented in the 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. Adekomaya, O.; Jamiru, T.; Sadiku, R.; Huan, Z. Sustaining the shelf life of fresh food in cold chain—A burden on the environment. *Alex. Eng. J.* **2016**, *55*, 1359–1365. [CrossRef]
2. Han, J.W.; Zuo, M.; Zhu, W.Y.; Zuo, J.H.; Lü, E.L.; Yang, X.T. A comprehensive review of cold chain logistics for fresh agricultural products: Current status, challenges, and future trends. *Trends Food Sci. Technol.* **2021**, *109*, 536–551. [CrossRef]
3. Huang, M.; Dong, L.; Kuang, H.; Jiang, Z.Z.; Lee, L.H.; Wang, X. Supply chain network design considering customer psychological behavior-a 4PL perspective. *Comput. Ind. Eng.* **2021**, *159*, 107484. [CrossRef]
4. Zhang, Y.; Chen, X. An optimization model for the vehicle routing problem in multi-product frozen food delivery. *J. Appl. Res. Technol.* **2014**, *12*, 239–250. [CrossRef]
5. Osvald, A.; Stirn, L.Z. A vehicle routing algorithm for the distribution of fresh vegetables and similar perishable food. *J. Food Eng.* **2008**, *85*, 285–295. [CrossRef]
6. Wang, S.; Sun, H.; Mou, J.; Jin, H. Optimization and efficiency of multi-temperature joint distribution of cold chain products: Comparative study based on cold accumulation mode and mechanical refrigeration mode. *J. Highw. Transp. Res. Dev.* **2016**, *33*, 146–153.
7. Wang, M.; Wang, Y.; Liu, W.; Ma, Y.; Xiang, L.; Yang, Y.; Li, X. How to achieve a win–win scenario between cost and customer satisfaction for cold chain logistics? *Phys. A Stat. Mech. Its Appl.* **2021**, *566*, 125637. [CrossRef]
8. Naderipour, M.; Alinaghian, M. Measurement, evaluation and minimization of CO_2, NOx, and CO emissions in the open time dependent vehicle routing problem. *Measurement* **2016**, *90*, 443–452. [CrossRef]
9. Liu, G.; Hu, J.; Yang, Y.; Xia, S.; Lim, M.K. Vehicle routing problem in cold Chain logistics: A joint distribution model with carbon trading mechanisms. *Resour. Conserv. Recycl.* **2020**, *156*, 104715. [CrossRef]
10. Wang, S.; Tao, F.; Shi, Y.; Wen, H. Optimization of vehicle routing problem with time windows for cold chain logistics based on carbon tax. *Sustainability* **2017**, *9*, 694. [CrossRef]

11. Bai, Q.; Yin, X.; Lim, M.K.; Dong, C. Low-carbon VRP for cold chain logistics considering real-time traffic conditions in the road network. *Ind. Manag. Data Syst.* **2022**, *122*, 521–543. [CrossRef]
12. Zhang, L.Y.; Tseng, M.L.; Wang, C.H.; Xiao, C.; Fei, T. Low-carbon cold chain logistics using ribonucleic acid-ant colony optimization algorithm. *J. Clean. Prod.* **2019**, *233*, 169–180. [CrossRef]
13. Yao, Q.; Zhu, S.; Li, Y. Green vehicle-routing problem of fresh agricultural products considering carbon emission. *Int. J. Environ. Res. Public Health* **2022**, *19*, 8675. [CrossRef]
14. Bao, H.; Fang, J.; Zhang, J.; Wang, C. Optimization on cold chain distribution routes considering carbon emissions based on improved ant colony algorithm. *J. Syst. Simul.* **2024**, *36*, 183–194.
15. Tao, N.; Yumeng, H.; Meng, F. Research on cold chain logistics optimization model considering low-carbon emissions. *Int. J.-Low-Carbon Technol.* **2023**, *18*, 354–366. [CrossRef]
16. Chen, J.K.C.; Yu, Y.W.; Batnasan, J. Services innovation impact to customer satisfaction and customer value enhancement in airport. In Proceedings of the Portland International Conference on Management of Engineering & Technology, Kanazawa, Japan, 27–31 July 2014.
17. Li, D.; Li, K. A multi-objective model for cold chain logistics considering customer satisfaction. *Alex. Eng. J.* **2023**, *67*, 513–523. [CrossRef]
18. Zhang, J.; Wang, W.; Zhao, Y.; Cattani, C. Multiobjective quantum evolutionary algorithm for the vehicle routing problem with customer satisfaction. *Math. Probl. Eng.* **2012**, *2012*, 879614. [CrossRef]
19. Xia, J.; He, Z.; Wang, S.; Liu, S.; Zhang, S. Mean–standard-deviation-based electric vehicle routing problem with time windows using Lagrangian relaxation and extended alternating direction method of multipliers-based decomposition algorithm. *Soft Comput.* **2024**, *28*, 7139–7160. [CrossRef]
20. Zhang, G.; Dai, L.; Yin, X.; Leng, L.; Chen, H. Optimization of multipath cold-chain logistics network. *Soft Comput.* **2023**, *27*, 18041–18059. [CrossRef]
21. Abbasi, M.; Rafiee, M.; Khosravi, M.R.; Jolfaei, A.; Menon, V.G.; Koushyar, J.M. An efficient parallel genetic algorithm solution for vehicle routing problem in cloud implementation of the intelligent transportation systems. *J. Cloud Comput.* **2020**, *9*, 1–14. [CrossRef]
22. 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. *Inf. Sci.* **2019**, *490*, 166–190. [CrossRef]
23. Shen, L.; Tao, F.; Wang, S. Multi-depot open vehicle routing problem with time windows based on carbon trading. *Int. J. Environ. Res. Public Health* **2018**, *15*, 2025. [CrossRef] [PubMed]
24. Zhang, L.; Fu, M.; Fei, T.; Lim, M.K.; Tseng, M.L. A cold chain logistics distribution optimization model: Beijing-Tianjin-Hebei region low-carbon site selection. *Ind. Manag. Data Syst.* **2024**. [CrossRef]
25. Cao, C.; Zhang, X.; Guo, Z. Vehicle routing problem with time windows arising in urban delivery. In Proceedings of the 4th International Conference on Electrical, Automation and Mechanical Engineering, Beijing, China, 21–22 June 2020; Volume 1626, p. 012097.
26. Qin, G.; Tao, F.; Li, L. A vehicle routing optimization problem for cold chain logistics considering customer satisfaction and carbon emissions. *Int. J. Environ. Res. Public Health* **2019**, *16*, 576. [CrossRef] [PubMed]
27. Liu, M. Five-elements cycle optimization algorithm for the travelling salesman problem. In Proceedings of the 2017 18th International Conference on Advanced Robotics (ICAR), IEEE, Hong Kong, China, 10–12 July 2017; pp. 595–601.
28. Ye, C.; Mao, Z.; Liu, M. A Novel Multi-Objective Five-Elements Cycle Optimization Algorithm. *Algorithms* **2019**, *12*, 244. [CrossRef]
29. Deb, K.; Agrawal, S.; Pratap, A.; Meyarivan, T. A fast elitist non-dominated sorting genetic algorithm for multi-objective optimization: NSGA-II. In Proceedings of the Parallel Problem Solving from Nature PPSN VI: 6th International Conference, Paris, France, 18–20 September 2000; Proceedings 6; Springer: Berlin/Heidelberg, Germany, 2000; pp. 849–858.
30. Xiaoning, W. Vehicles' Distribution Route Research of Fresh Agricultural Products of Cold Chain Logistics under the Agricultural Super-Docking Mode. Master's Thesis, East China Jiaotong University, Nanchang, China, 2014.
31. Xiang, Y.; Guo, J.; Jiang, C.; Ma, H.; Liu, M. Multi-Objective Five-Element Cycle Optimization Algorithm Based on Multi-Strategy Fusion for the Bi-Objective Traveling Thief Problem. *Appl. Sci.* **2024**, *14*, 7468. [CrossRef]
32. Czyżżak, P.; Jaszkiewicz, A. Pareto simulated annealing–A metaheuristic technique for multiple-objective combinatorial optimization. *J.-Multi-Criteria Decis. Anal.* **1998**, *7*, 34–47. [CrossRef]
33. Zitzler, E.; Thiele, L.; Laumanns, M.; Fonseca, C.M.; Da Fonseca, V.G. Performance assessment of multiobjective optimizers: An analysis and review. *IEEE Trans. Evol. Comput.* **2003**, *7*, 117–132. [CrossRef]
34. Van Veldhuizen, D.A.; Lamont, G.B. Evolutionary computation and convergence to a pareto front. In Proceedings of the Late Breaking Papers at the Genetic Programming 1998 Conference, Madison, WI, USA, 22–25 July 1998; pp. 221–228.
35. Lu, C.; Xiao, S.; Li, X.; Gao, L. An effective multi-objective discrete grey wolf optimizer for a real-world scheduling problem in welding production. *Adv. Eng. Softw.* **2016**, *99*, 161–176. [CrossRef]
36. Kumawat, I.R.; Nanda, S.J.; Maddila, R.K. Multi-objective whale optimization. In Proceedings of the Tencon 2017–2017 IEEE Region 10 Conference, IEEE, Penang, Malaysia, 5–8 November 2017; pp. 2747–2752.
37. Papasani, A.; Devarakonda, N. A novel feature selection algorithm using multi-objective improved honey badger algorithm and strength pareto evolutionary algorithm-II. *J. Eng. Res.* **2022**, *11*, *2B*. [CrossRef]

38. Ali, M.H.; Salawudeen, A.T.; Kamel, S.; Salau, H.B.; Habil, M.; Shouran, M. Single-and multi-objective modified aquila optimizer for optimal multiple renewable energy resources in distribution network. *Mathematics* **2022**, *10*, 2129. [CrossRef]
39. Zeleny, M. *Multiple Criteria Decision Making*; McGraw-Hill Company: New York, NY, USA, 1982.