applied sciences

MDPI

Article

A Deep Reinforcement Learning-Based Algorithm for Multi-Objective Agricultural Site Selection and Logistics Optimization Problem

Huan Liu *, Jizhe Zhang, Zhao Zhou, Yongqiang Dai and Lijing Qin

College of Information Science and Technology, Gansu Agricultural University, Anning, Lanzhou 730070, China; zhangjizhe402@gmail.com (J.Z.); maederj755@gmail.com (Z.Z.); daiyq@gsau.edu.cn (Y.D.); qinlj@gsau.edu.cn (L.Q.)

* Correspondence: liuh@gsau.edu.cn

Abstract: The challenge of optimizing the distribution path for location logistics in the cold chain warehousing of fresh agricultural products presents a significant research avenue in managing the logistics of agricultural products. The goal of this issue is to identify the optimal location and distribution path for warehouse centers to optimize various objectives. When deciding on the optimal location for a warehousing center, various elements like market needs, supply chain infrastructure, transport expenses, and delivery period are typically taken into account. Regarding the routes for delivery, efficient routes aim to address issues like shortening the overall driving distance, shortened travel time, and preventing traffic jams. Targeting the complex issue of optimizing the distribution path for fresh agricultural products in cold chain warehousing locations, a blend of this optimization challenge was formulated, considering factors like the maximum travel distance for new energy trucks, the load capacity of the vehicle, and the timeframe. The Location-Route Problem with Time Windows (LRPTWs) Mathematical Model thoroughly fine-tunes three key goals. These include minimizing the overall cost of distribution, reducing carbon emissions, and mitigating the depletion of fresh agricultural goods. This study introduces a complex swarm intelligence optimization algorithm (MODRL-SIA), rooted in deep reinforcement learning, as a solution to this issue. Acting as the decision-maker, the agent processes environmental conditions and chooses the optimal course of action in the pool to alter the environment and achieve environmental benefits. The MODRL-SIA algorithm merges a trained agent with a swarm intelligence algorithm, substituting the initial algorithm for decision-making processes, thereby enhancing its optimization efficiency and precision. Create a test scenario that mirrors the real situation and perform tests using the comparative algorithm. The experimental findings indicate that the suggested MODRL-SIA algorithm outperforms other algorithms in every computational instance, further confirming its efficacy in lowering overall distribution expenses, carbon emissions, and the depletion of fresh produce in the supply chain of fresh agricultural products.

Keywords: deep reinforcement learning; Vehicle Routing Problem; location problem; swarm intelligence optimization algorithm; fresh agricultural product supply chain

Citation: Liu, H.; Zhang, J.; Zhou, Z.; Dai, Y.; Qin, L. A Deep Reinforcement Learning-Based Algorithm for Multi-Objective Agricultural Site Selection and Logistics Optimization Problem. *Appl. Sci.* **2024**, *14*, 8479. https://doi.org/10.3390/app14188479

Academic Editor: Arkadiusz Gola

Received: 11 June 2024
Revised: 10 September 2024
Accepted: 16 September 2024
Published: 20 September 2024

1. Introduction

Fresh agricultural products are essential for daily life and constitute a significant market sector [1]. The logistics of these products involve a dynamic circulation process that encompasses early production, processing, integrated packaging, warehouse storage, and subsequent distribution to consumers, accompanied by continuous information exchange [2]. However, the logistics chain for fresh produce faces various risks during implementation, including costs and potential damage to the goods. Therefore, it is imperative to adopt cold chain transportation and distribution solutions to improve the current situation.

Fresh agricultural products are characterized by their variety, short life cycles, perishability, high logistics demands, substantial volume, and stringent storage requirements [3]. Consequently, the logistics service quality standards for these products are distinct from those for other goods, necessitating elevated service levels. Furthermore, the primary consumption areas for these products are spread across various urban locations. The need to deliver to numerous customer points, coupled with uneven distribution requirements, complicates the delivery process. In the traditional supply chain, fresh agricultural products pass through several stages, including distributors and retailers, before reaching consumers [4]. However, in modern supply chain systems, the process is streamlined to just three levels: producer, fresh agricultural product platform, and consumer. This simplification significantly reduces both the intermediary cost and the time needed for purchasing.

Establishing a cold chain distribution system for fresh agricultural products requires meticulous attention to proper storage and distribution from platform warehouses to consumers. Representing the agricultural aspect of Industry 4.0, modern agriculture differs from agricultural industrialization by embracing smart agriculture. This approach is characterized by automation, personalization, artistic integration, ecological considerations, large-scale operations, and precision agriculture, all driven by the smart economy with a primary focus on the big health industry [5]. Ensuring the timely delivery of agricultural products is crucial to maintaining freshness and minimizing losses due to improper storage. In accordance with the central government's "double carbon" policy, the newly established logistics and distribution system employs new energy vehicles, thereby setting higher standards for cold chain distribution and transportation solutions [6].

The main contributions of this article are summarized as follows:

(1) This paper is the first to construct a mathematical model of the Location-Routing Problem with Time Windows (LRPTWs) and introduces a multi-objective framework for LRPTW based on three distinct objectives.
(2) A multi-objective swarm intelligence algorithm framework utilizing deep reinforcement learning was developed to minimize operating costs, emissions, and product losses. The DQNAgent module, trained with optimized network parameters via reward and punishment functions, incorporates various local search strategies to enhance solution quality.
(3) The effectiveness of the MODRL-SIA in addressing the multi-objective LRPTW problem was demonstrated by superior performance metrics, specifically through Hypervolume (HV) and Inverted Generational Distance (IGD) indicators.

The remaining structure of this paper is as follows: Section 2 reviews the relevant literature, Section 3 outlines the definition of the LRPTW problem, and Section 4 introduces the details of the MODRL-SIA algorithm framework. Section 5 describes the experiments and results analysis. The final section summarizes the entire paper, discussing its impact and limitations.

2. Related Work

To ensure the delivery of fresh agricultural products to consumers with enhanced quality and to align with the "double carbon" policy while reducing distribution costs, researchers have implemented various optimization strategies. These strategies focus on minimizing delivery distance, time, costs, and carbon emissions. In fundamental research, the distribution challenges of fresh agricultural products within a region are modeled as a Location-Routing Problem (LRP). This model takes into account factors such as vehicle load constraints, travel distance limitations, and time window constraints at customer points and distribution warehouses.

Maghfiroh, M.F.N. et al. [7] suggest integrating Variable Neighborhood Search (VNS) with Path Relinking (PR) to address the LRPTW, considering its extensive solution pools. Yan, X.M. et al. [8] have designed an evolutionary multitasking optimization algorithm to simultaneously solve multiple routing tasks. Hassanpour, S. T. et al. [9] developed a Mixed-Integer Linear Programming (MILP) model for the LRPTW. Sutrisno, H. et al. [10]

propose the CSNS, which considers a probabilistic mechanism for facility selection and utilizes the k-means clustering algorithm to generate routing solutions with simultaneous neighborhood search. Tadaros, M. et al. [11] summarize and discuss the multi-objective LRP, demonstrating its relevance to practical application environments. They introduce improved methods for solving multi-objective LRP, including a weighted summing technique and a Genetic Algorithm (GA) [12]. Additional Developments: The uncertainty of some key parameters in the LRP is addressed through the application of a robust fuzzy optimization model [13]. Furthermore, fuzzy chance-constrained programming is proposed to tackle demand uncertainty and resolve the uncertainty problem [14].

Domestic and international scholars have extensively explored the LRP in terms of model construction, extraction of real production constraints, and development of scheduling solution algorithms. The LRP integrates vehicle routing and location problems, both recognized as NP-hard, which explains the relative scarcity of research in this field. Notably, to ensure the timely delivery of fresh agricultural products, time window constraints are incorporated into the LRP. Traditionally, LRP research often omits environmental facility constraints to simplify the mathematical model and ease the difficulty of solving it. However, real-world constraints are crucial in actual distribution environments. For example, new energy trucks face energy limitations, and delivery points must operate within specific time windows. Consequently, studying the multi-constrained LRPTW is of significant practical research value.

Domestic and foreign scholars have extensively explored and researched LRP in terms of model construction, actual production constraint extraction, and scheduling solution algorithms. LRP combines the vehicle route planning problem and location problem. Since both are NP-hard problems, there are relatively few research results in this area. Particularly, to ensure the timely delivery of fresh agricultural products, time window constraints are added to the LRP problem. In existing LRP research, environmental facilities are typically not constrained to simplify the mathematical model and reduce solving difficulty. However, real-world constraints need consideration in actual distribution environments. For instance, new energy trucks have energy restrictions, and delivery points operate within specified time windows. Therefore, studying the multi-constrained LRPTW holds significant practical research value.

Kallestad, J. et al. [15] have developed a general framework for solving various combinatorial optimization problems by integrating deep reinforcement learning (RL) agents into the Adaptive Large Neighborhood Search (ALNS) framework. Rotaeche, R. et al. [16] propose the application of deep reinforcement learning (DRL) to the task allocation problem in Critically Adaptive Distributed Embedded Systems (CADES). Fang, J. et al. [17] introduce a DRL-based algorithm to solve the one-dimensional cutting stock problem (1-DCSP). Tu, C.F. et al. [18] suggest a new general framework for online packaging problems using deep reinforcement learning hyperheuristics.

The LRPTW model presents a relatively complex problem space due to variations in solution models. Deep reinforcement learning can elucidate the structural characteristics of the LRPTW model by constructing a probabilistic model, which is implicitly utilized during the training process. These algorithms autonomously search the potential problem space and have proven effective in function optimization and combinatorial optimization challenges, including the hybrid flow shop scheduling problem (HFSP) [19] and the Boolean satisfiability problem (SAT) [20]. Given their practical research value across various domains, this paper employs a deep reinforcement learning algorithm framework combined with the actual LRPTW model to tackle the real cold chain distribution problem of fresh agricultural products. The LRPTW reality model is depicted in Figure 1, which uses Google Satellite Maps as its source.

Real-world problems often entail multiple objectives, driving the need to enhance algorithms into multi-objective formats. Jiang, S. Y. et al. [21] conducted an extensive survey and classification of existing research in Evolutionary Dynamic Multi-Objective Optimization (EDMO). Khalid, A.M. et al. [22] used IGD and HV indicators to validate

their multi-objective algorithm. Shu, X.L. et al. [23] proposed a Multi-Objective Particle Swarm Optimization (D-MOPSO) with a dynamic population size to tackle the issues of convergence and diversity inherent in particle swarm optimization. Pirouz, B. et al. [24] utilized Sparse Optimization for feature selection in Linear SVM, applying it to Multi-Objective Optimization. Yang, Y.F. et al. [25] employed the e-constraint method to adjust the e-value at different stages to meet the demands of their multi-objective algorithm.

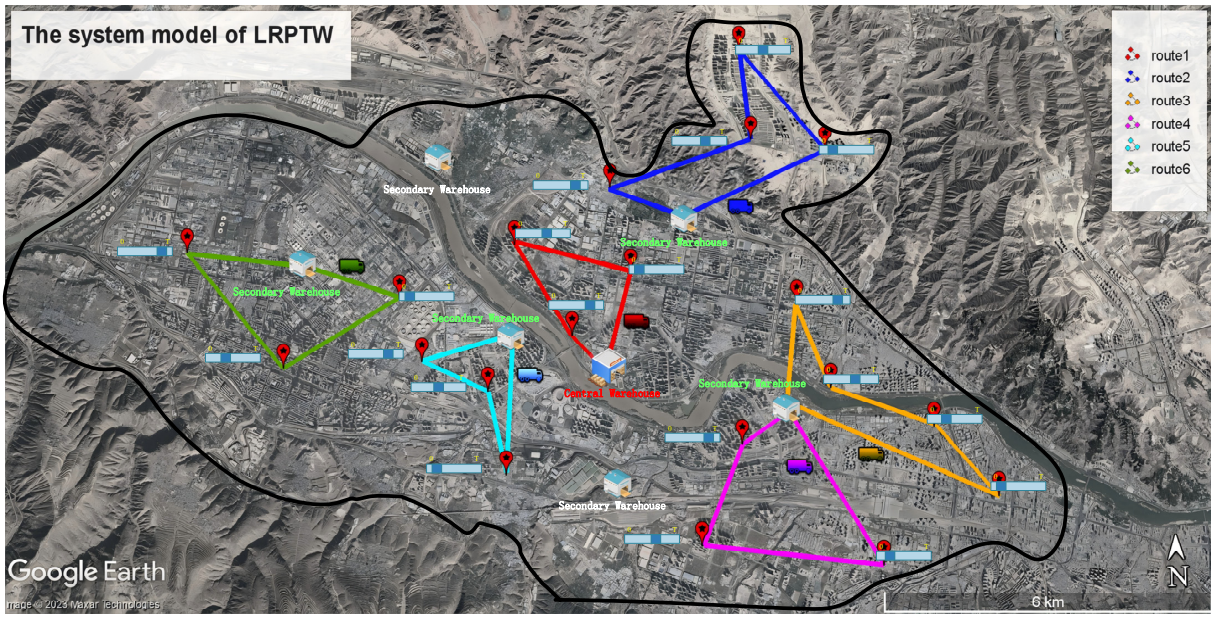

Figure 1. A realistic model of the LRPTW problem.

3. LRPTW Model

The problem model studied in this article originates from the order distribution of fresh agricultural products on an e-commerce platform. The distribution process can be abstracted into central warehouses, auxiliary warehouses, customer points, and distribution vehicles. The distribution warehouses can choose a central warehouse and an uncertain number of auxiliary warehouses. Within the distribution warehouse, multiple vehicles can be used simultaneously. Due to the construction cost of the auxiliary warehouses, time window limitations at customer points, and load and energy limitations of the distribution vehicles, the path planning problem can be abstracted as LRPTW.

In LRPTW, there are distribution warehouses and receiving points. Vehicles departing from the distribution warehouses eventually return to these distribution warehouses. The vehicle parameters of the distribution warehouses are the same, but the time window at customer points varies. Reasonable assumptions can simplify the model construction process, reduce the consideration of secondary factors, and decrease the difficulty of model construction. Therefore, to control some unexplained variables and directly reflect the basic problems of the model, this article simplifies the actual situation. It proposes conditions for an appropriate environment required for model building.

Customer points: To meet the needs of each customer point, only one delivery vehicle is allowed to serve it and visit it only once. Additionally, the demand at any receiving point must be less than the maximum load of the vehicle.

Delivery vehicles: All delivery vehicles are of the same type with identical load and speed.

Delivery routes: Based on demand, the distribution center has enough vehicles to deliver each route on time. Transportation is one-way from the distribution center to the receiving point.

Distribution warehouses: The start and final destination of each delivery vehicle are the same distribution warehouse, with no midway allocation.

Load Limit: The total demand from all customers on the same route must not exceed the rated load of the vehicle.

Energy limitations: Each delivery vehicle needs to return to the distribution center for energy replenishment, and the one-way distance cannot exceed the maximum driving distance.

Time window restriction: The time window should at least meet the condition that the vehicle path contains only one demand node. Delivery vehicle arrival times and loading operations should occur within the acceptable time window at the delivery point.

Other settings: The delivery process does not consider road congestion, adverse weather conditions, or vehicle breakdowns.

The definitions of mathematical symbols used in the problem model are shown in Table 1. In the multi-objective location-route problem, the three objectives are the minimum cost objective, the minimum carbon emission objective, and the minimum product loss objective. The mixed-integer programming model of this problem is as follows:

Table 1. Mathematical symbols.

Symbol	Practical Significance
n	Number of vehicles
m	Number of receiving points
a	Number of new warehouses
k	Vehicle number $k \in \{1, 2, \ldots, n\}$
i, j, q	Receiving point number
p	Distribution center number
f_1	The cost of each new warehouse
f_2	Cost per vehicle per kilometer
f_3	Fixed cost per vehicle
f_4	Cooling mass produced per minute
f_5	Mass produced by fuel consumption per kilometer driven
f_6	Product loss coefficient
D_{ij}	The distance between the delivery point and
T_{ij}	The time between the delivery point and
X_{ik}	0.1 decision variable, indicating that the first vehicle completes the delivery operation at the receiving point
X_{ijk}	0.1 decision variable, indicating that the first vehicle is the receiving point and the delivery operation is completed.
X_{ipk}	0.1 decision variable, indicating that the first vehicle completes the delivery operation for the distribution center and receiving point
Q_k	Actual load of the k-th vehicle
Q_i	Required weight at delivery point i
Q	Delivery vehicle maximum load
L_k	Actual distance traveled by the k-th vehicle
L	Maximum driving distance of delivery vehicles
e_i	Earliest delivery time at delivery point i
e_p	Delivery center p earliest delivery time
l_i	Latest delivery time at delivery point i
l_p	Latest delivery time for distribution center p
A_{ik}	Actual delivery time at delivery point i
A_{pk}	Actual delivery time of distribution center p

Objective:

$$\mathrm{MinCost} = \mathrm{Cost}_1 + \mathrm{Cost}_2 + \mathrm{Cost}_3 \tag{1}$$

$$\mathrm{MinCO}_2 = \mathrm{CO}_{21} + \mathrm{CO}_{22} \tag{2}$$

$$\mathrm{MinLoss} = \mathrm{Loss} \tag{3}$$

W.R.T:

$$Cost_1 = a \times f_1 \tag{4}$$

$$Cost_2 = \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n} D_{ij} \times f_2 \times X_{ijk} \tag{5}$$

$$Cost_3 = n \times f_3 \tag{6}$$

$$CO_{21} = \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n} T_{ij} \times f_4 \times X_{ijk} \tag{7}$$

$$CO_{22} = \sum_{i=1}^{m}\sum_{j=1}^{m}\sum_{k=1}^{n} D_{ij} \times f_5 \times X_{ijk} \tag{8}$$

$$Loss = \sum_{k=1}^{n} Q_k \times f_6 \tag{9}$$

$Cost_1$ is the cost of a new warehouse, $Cost_2$ is the fuel cost, $Cost_3$ is the vehicle fixed cost, CO_{21} is the refrigeration carbon emissions, CO_{22} is the fuel consumption carbon emissions, and $Loss$ is the product loss.

Subject to:

$$X_{ijk} = \begin{cases} 1, \text{the delivery operation of customer i} \\ \text{and j are completed by the } k_{th} \text{ vehicle} \\ 0, \text{others} \end{cases} \tag{10}$$

$$X_{ik} = \begin{cases} 1, \text{the delivery operation of customer} \\ \text{i is carried out by the } k_{th} \text{ vehicle} \\ 0, \text{others} \end{cases} \tag{11}$$

$$\sum_{i=1}^{m}\sum_{k=1}^{n} X_{ipk} = \sum_{i=1}^{m}\sum_{k=1}^{n} X_{pik} \tag{12}$$

$$\sum_{i=1}^{m} X_{ijk} = \sum_{i=1}^{m} X_{jik} (k \in n, \forall j \in m) \tag{13}$$

$$Q_k \leq Q \tag{14}$$

$$\sum_{k=1}^{n} X_{ik} = 1 (\forall i \in m) \tag{15}$$

$$Q \geq Q_i (\forall i \in m) \tag{16}$$

$$\sum_{i=1}^{m} X_{iqk} = \sum_{j=1}^{m} X_{qjk} (k \in n, \forall q \in m) \tag{17}$$

$$e_i \leq A_{ik} \leq l_i \tag{18}$$

$$e_p \leq A_{pk} \leq l_p \tag{19}$$

$$L_k = \sum_{i=1}^{m} X_{ijk} \times D_{ij} (k \in n, \forall j \in m) \tag{20}$$

$$L_k \leq L \tag{21}$$

In this model, Equation (1) represents the cost objective function of the multi-objective Location-Routing Problem, aiming to minimize the distribution cost while satisfying the constraints. Equation (2) represents the carbon dioxide emission objective function, aiming to minimize carbon dioxide emissions during the distribution process while satisfying the constraints. Equation (3) represents the product loss objective function, aiming to minimize the loss of fresh agricultural products during the distribution process while satisfying the constraints. Equation (4) represents the cost of building a new warehouse during the distribution process, obtained by multiplying the number of new warehouses selected and the cost of building each new warehouse. Equation (5) represents part of the cost

incurred along with mileage and fuel consumption during transportation, also known as the variable cost per vehicle kilometer, obtained by multiplying the mileage of the delivery vehicle and the transportation cost per kilometer of distance. Equation (6) represents fixed costs that do not depend on the distance traveled by the vehicle or the quantity of goods transported, including administrative expenses, repair and maintenance expenses, road maintenance fees, bridge fees, and other miscellaneous expenses. This part of the cost is only related to the number of vehicles. Equation (7) represents the carbon emissions caused by refrigeration during the vehicle's travel time, which is only related to the vehicle's travel time. Equation (8) represents the carbon emissions caused by fuel consumption related to the distance traveled by the vehicle, which is only related to the distance traveled by the vehicle. Equation (9) represents the loss caused by the distance traveled by the vehicle, where the loss is related to the vehicle load and loss coefficient. Equations (10) and (11) represent decisions made through 0–1 variables. Equation (12) indicates that all vehicles depart from the distribution center and return to it after completing all distribution tasks. Equation (13) indicates that the number of incoming and outgoing vehicles at each receiving point is equal. Equations (14) and (15) indicate that the load of each vehicle cannot exceed its maximum load limit. Equation (16) indicates that each receiving point only allows one delivery vehicle to serve it and visit it once. Equation (17) indicates that the maximum load of the vehicle must meet the demand at any receiving point. Equation (18) indicates that after completing a delivery task, the delivery vehicle must proceed to the next delivery point. Equation (19) represents time window constraints that must be satisfied for all receiving points. Equation (20) indicates that the time window constraints for all distribution centers must be satisfied. Equation (21) indicates that the driving distance of each vehicle cannot exceed its maximum driving distance.

In summary, the multi-objective LRPTW model integrates equations designed to minimize distribution costs, carbon dioxide emissions, and product losses while complying with a variety of constraints. These equations account for factors such as the construction of new warehouses, variable transportation costs, fixed expenses, emissions from refrigeration and fuel consumption, and losses incurred during transit. Additionally, the model considers decision variables, vehicle departure and arrival conditions, load limits, and time window constraints to optimize distribution efficiency.

The LRPTW model addresses three interrelated subproblems: coordinating objectives, optimizing vehicle routes, and ensuring adherence to constraints. The optimization of vehicle routes is crucial, as it directly affects the satisfaction of constraints and determines the viability of the routes. To address these subproblems effectively, a collaborative approach is essential, incorporating their characteristics into the MODRL-SIA framework to produce high-quality solutions. By collectively considering these subproblems and leveraging the MODRL-SIA framework, the model efficiently generates optimal solutions that balance cost, environmental sustainability, and operational efficiency in distribution logistics.

4. MODRL-SIA Algorithm

4.1. MODRL-SIA Algorithm Framework

The MODRL-SIA algorithm utilizes the water wave optimization algorithm as its fundamental framework. The trained DQNAgent model acts as the decision maker, and together, they form the MODRL-SIA framework model, as illustrated in Figure 2. The algorithmic process of the MODRL-SIA framework is described in Algorithm 1. After being trained for a specific number of iterations, the MODRL-SIA framework algorithm can effectively address similar types of problems. It can serve as a versatile method for solving various combinatorial optimization problems.

Algorithm 1: MODRL-SIA algorithm framework process

Input: wavelength λ, wave height h, breaking wave coefficient a, and maximum number of iterations t_{max}
While $t < t_{max}$ **do**
  For i **in** Pop **do**
    DQNAgent selects low-energy consumption strategy.
    If $f(x') < f(x^*)$ **then**
      DQNAgent selects high-energy consumption strategy.
    End If
    If $h = 0$ **then**
      DQNAgent selects a true random strategy.
    End If
  End For
End While

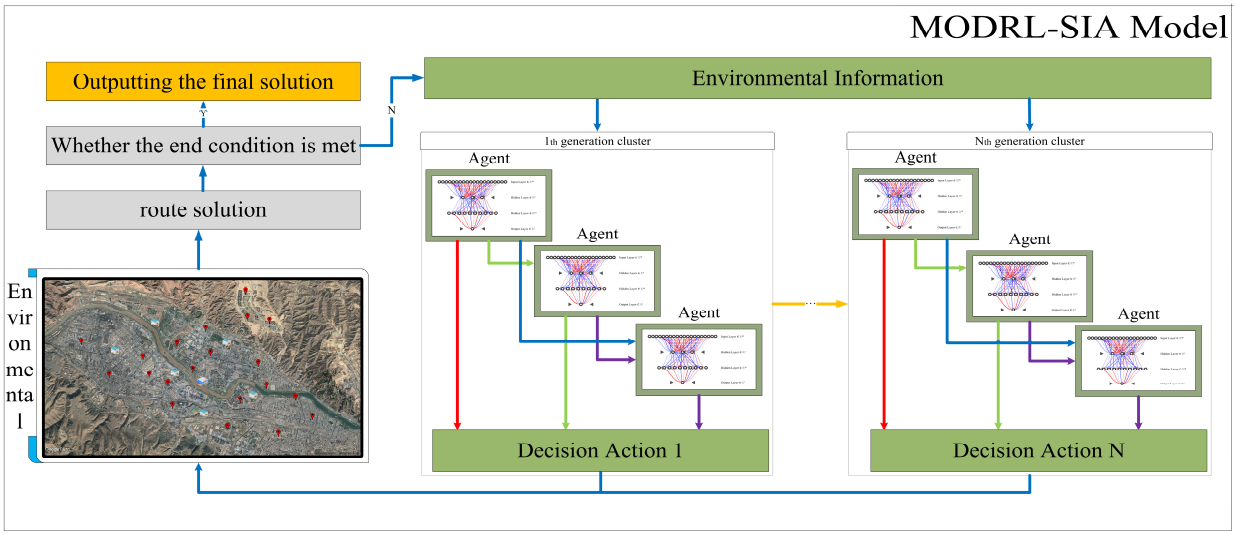

Figure 2. MODRL-SIA framework model.

4.2. Deep Reinforcement Learning

Deep reinforcement learning is more akin to a human's mode of learning new things. Targeting the task and environment, it ultimately accomplishes the specified task by continuously learning from mistakes and refining strategies. Classic deep reinforcement learning comprises two crucial components, namely, the environment and the agent. The agent serves as the decision-maker, taking the environmental state as input and predicting the best action to modify the environment and obtain environmental rewards. The multi-objective location-routing problem is an NP-hard challenge. Additionally, the locations, quantities, and delivery times of receiving points are subject to constant change, rendering the solution more intricate [26].

Deep learning is currently a prominent field of artificial intelligence research. It operates as a data-driven machine learning technique, predicting the likelihood of future events based on existing historical data. While relatively mechanistic and static, it proves unsuitable for dynamic vehicle route scheduling in agricultural supply distribution. Reinforcement learning involves inferring the probability of an action occurring at the next moment based on the current and previous states and actions. It represents a continuously changing process enabling vehicle routing decisions. However, due to the complexity of vehicle paths and the continuous state changes, the discretized state space becomes extensive. Traditional reinforcement learning methods, such as Q-Learning, struggle to maintain a vast Q table in memory [27].

Deep reinforcement learning uses neural networks instead of Q tables and experience replay mechanisms to address training sample issues. Combining the decision-making ability of reinforcement learning with the perception ability of deep learning, it provides an effective approach to resolving complex perception decision-making problems. Optimizing discrete combinations using the randomness of swarm intelligence algorithms combined with deep reinforcement learning's perceptual capabilities yields enhanced solution accuracy, faster convergence speeds, and greater resilience to local optimality. DQN (Deep Q Network) [28] is the most commonly used deep reinforcement learning algorithm. We will employ the MODRL-SIA framework to tackle this issue. The MODRL-SIA algorithm integrates the trained agent with swarm intelligence algorithms, utilizing them to replace the original decision-making algorithm and enhance its optimization speed. The following section outlines the representation method of the DQNAgent's state space, action space, and reward and punishment function in the MODRL-SIA framework.

State Space: The goal of this problem is to minimize cold chain distribution costs, minimize carbon emission indicators, and minimize product losses. Simultaneously, it is considered that dividing the receiving point into different vehicles will cause changes in the vehicle path, thus affecting the objective. Therefore, the state of the vehicle path is formally represented as the state of the environment. It mainly consists of three parts: distribution center information, receiving point information, and vehicle information.

Action space: The task of the path planning allocator is to assign each order to the appropriate path. Assuming that the number of receiving points is m, its action space is expressed as $A = \{P_1,\ P_2, P_3,\ \cdots, P_i,\ \cdots,\ P_m\}$. Indicates the sequence code of all receiving points. Then, the sequence code is cut according to the constraints. For example, the first complete vehicle planning path after cutting is $[d_1, 28, 15, 21, 14, 8, d_1]$. The meaning is: Distribution center d_1>>Receipt point 28>>Receipt point 15>>Receipt point 21>>Receipt point 14>>Receipt point 8>>Distribution center d_1. In the same way, other routes will allocate vehicles for delivery in the order of sequence codes.

Reward and Punishment Function: The design of reward and punishment functions is an extremely important part of deep reinforcement learning. By specifying and quantifying the task goals, the DQNAgent is guided to select an action strategy from the action pool. Whether the design of the reward function meets the target requirements will determine whether the DQNAgent can learn the desired strategy, and this affects the convergence speed and final performance of the algorithm. Given that the optimization goal of the current stage is to minimize distribution costs, if action strategies with poor effects are often chosen, it will inevitably result in the inability to find the optimal solution at the end of the iteration. Therefore, the reward and punishment function is defined as follows:

$$R = (1 + \alpha) \times R \tag{22}$$

$$R = \beta \times R \tag{23}$$

Equations (22) and (23) represent the reward function and penalty function. In these equations, R represents the probability of selecting a policy from the policy pool, α represents the learning rate of the DQNAgent, and β represents the discount rate of the DQNAgent.

4.3. Swarm Intelligence Optimization Algorithm

Among the swarm intelligence optimization algorithms, the water wave optimization algorithm [29,30] has the advantages of fewer control parameters, simple operation, and easy implementation. It has achieved great success in target optimization, engineering calculations, combination optimization, etc. In the water wave optimization algorithm, three forms of water waves are employed to explore the search space: propagation operation, refraction operation, and wave breaking operation. The propagation operation involves the spreading of water waves; refraction changes the direction of propagation when the height of the water wave reaches 0, and breaking waves occur when the water

wave height is too high, leading to the formation of sprays. The DQNAgent module is trained to select action strategies in the action pool based on the actual problem and its subproblems. The trained DQNAgent model is then used in the water wave optimization algorithm to perform optimization operations among the three operations.

4.4. Decision-Making Action Design of Action Pool

The decision-making design of action pools is divided into two categories: random decision-making strategies and purposeful decision-making strategies. Random decision-making strategies are capable of escaping local optima, while purposeful decision-making strategies can accelerate convergence. This article designs a total of 10 decision-making strategies. Random decision-making strategies include full random, random regret, random greedy, and difference-seeking random repair large neighborhood search. Purposeful decision-making strategies include self-learning search strategy, optimal learning search strategy, near-neighbor search strategy, variable neighborhood search strategy, difference-seeking regretful, and difference-seeking greedy large neighborhood search strategies.

4.5. DQNAgent Training

The DQNAgent model uses the action pool to fairly select decision-making strategies in the initial situation. Decision-making strategies are selected through the neural network. After each action is completed, the current status and reward and punishment values are fed back to the DQNAgent model based on the results obtained. The DQNAgent model is illustrated in Figure 3. The learning and training process of the DQNAgent in each round is shown in Algorithm 2.

Algorithm 2: DQNAgent learning and training process

Input: Q-Learning Tbale, State Space, and maximum number of iterations t_{max}
Output: Q-Learning Tbale, action space
While $t < t_{max}$ **do**
  The selected decision actions are used to handle candidate solutions.
  Calculate the rewards and penalties obtained from this processing.
  Calculate the current environment state.
  Save the current status and reward and punishment values to the memory pool.
End While

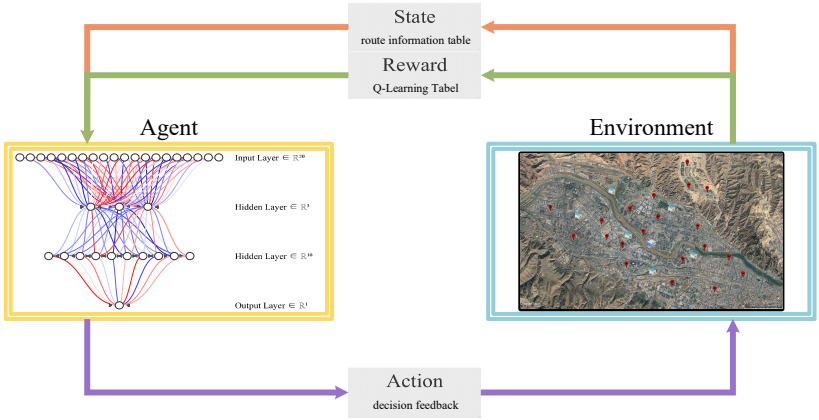

Figure 3. DQNAgent model.

5. Experimental Results and Analysis

The experimental environment uses the Python3.8 language environment (Anaconda3). The operating system is Windows 10. The configuration of the experimental machine is a Core i5CPU with a main frequency of 3.6 Ghz.

This paper has designed a total of four scale scenarios, including small-scale retail models, medium-scale town models, large-scale regional clusters, and extra-large-scale urban models. Each scenario consists of 10 variants, with each variant being run 10 times. The termination condition for the experiments is time-controlled, ensuring that each test is uniformly evaluated under similar temporal constraints. The termination condition during the experiment was set to reaching the predetermined running time. Verify the performance of the proposed MODRL-SIA framework algorithm in solving LRPTW. Customer points are randomly distributed on a 100×100 km map, and the central warehouse coordinates are (50, 50). Optional auxiliary warehouses are distributed at (25, 25), (75, 75), (25, 75), and (75, 25) with a construction cost of 1000 yuan each. The size of customer points is divided into small-scale (30 customer points), medium-scale (50 customer points), large-scale (80 customer points), and ultra-large (100 customer points). The number of central warehouses is set to one, and the number of optional auxiliary warehouses is set to four. The time window length of the customer point is set to 100 min, and the start time of the time window of the customer point is set to a random number between 0 and 1340. The demand at the customer point is set to a random number between 1 and 100 kg, the unloading time at the customer point is set to 30 min, and the time window of the warehouse is set to 0–1440 min. Taking a 4.2 m refrigerated truck as an example, fuel consumption is 15 L per 100 km, the gasoline cost is 8 yuan per L, refrigeration emits 0.2 kg of $C0_2$ per hour, and gasoline combustion emits 2.254 kg of $C0_2$ per liter. The vehicle traveling speed is 0.83 km/min, the vehicle fuel cost is 1.2 yuan/km, and the vehicle fixed cost is 100 yuan. The maximum load capacity of the vehicle is 1000 kg, and the maximum driving distance of the vehicle is set to 1000 km. Transportation losses occur when a single vehicle trip is greater than 100 km, at 5% when less than 200 km, 10% when less than 300 km, 15% when less than 400 km, 20% when less than 500 km, and 30% when more than 500 km. According to the commonly used means of transport and their properties in reality, the details of test examples are shown in Table 2.

Table 2. Example tables for different sizes.

Example No.	Number of Customer Points	Number of Centralized Warehouses	Number of Optional Warehouses	Length of Time Window	Time Window Range	Customer Point Demand	Vehicle Speed	Vehicle Fuel Costs	Vehicle Fixed Costs	Unloading Times	Maximum Vehicle Load	Maximum Vehicle Distance
Example 1	30	1	4	100 min	0–1440 min	1–100 kg	0.83 km/min	1.2/km	100	30 min	1000 kg	1000 km
Example 2	50	1	4	100 min	0–1440 min	1–100 kg	0.83 km/min	1.2/km	100	30 min	1000 kg	1000 km
Example 3	80	1	4	100 min	0–1440 min	1–100 kg	0.83 km/min	1.2/km	100	30 min	1000 kg	1000 km
Example 4	100	1	4	100 min	0–1440 min	1–100 kg	0.83 km/min	1.2/km	100	30 min	1000 kg	1000 km

5.1. Experimental Analysis of ALNSWWO and Comparative Algorithms

The parameters of the deep reinforcement learning network model include model parameters and hyperparameters. Model parameters are often adjusted driven by data, such as the specific kernel parameters of the convolution kernel and the weights of the neural network. Hyperparameters, on the other hand, do not need to be driven by data but are manually adjusted before or during training. For example, the learning rate and discount rate are hyperparameters. Hyperparameters are usually initially set based on experience and then tuned based on the training effect. The sparse reward in the algorithm is determined by the learning rate and discount rate, which directly control the magnitude of network gradient updates during training. Both the learning rate and discount rate are set at four levels based on experience. The detailed design of the orthogonal experiment is shown in Table 3.

Table 3. Orthogonal experiment details.

Experiment No.	Learning Rate (α)	Discount Rate (β)
1	0.001	0.95
2	0.001	0.9
3	0.001	0.85
4	0.001	0.7
5	0.01	0.95
6	0.01	0.9
7	0.01	0.85
8	0.01	0.7
9	0.05	0.95
10	0.05	0.9
11	0.05	0.85
12	0.05	0.7
13	0.1	0.95
14	0.1	0.9
15	0.1	0.85
16	0.1	0.7

Each set of experiments was trained for 5000 s, and the 16 sets of training results were combined into a collection. Obtain a set of optimal non-dominated solutions through non-dominated algorithms. A total of 88 non-dominated solutions were obtained, with the largest proportion being 38 in Orthogonal Experiment 5. The Pareto front of this solution set is shown in Figure 4. Therefore, the algorithm hyperparameters are set to α equals 0.01 and β equals 0.95, respectively.

5.2. Comparison Experiments

The comparison algorithm selects the currently better multi-objective hybrid algorithms MOPSO-GA and MOGA-VNS. Using the same time, HV and IGD were used as experimental evaluation indicators for comparison. Test the solution performance of the algorithm proposed in this article.

HV is used to evaluate the degree to which the target space is covered by an approximation set and is the most common evaluation index. A reference point is required. The three targets are normalized separately, and the reference point is set to (1.2, 1.2, 1.2). The HV value is the volume of the hypercube formed between the Pereto front (PF) and the reference point. The comparison of HV does not require a priori knowledge and does not require finding the true Pareto front. If a certain approximation set A completely dominates another approximation set B, then the supercapacity HV of A will be greater than that of B. Therefore, HV can be completely used for PF comparison. The HV calculation method is shown in Equation (24).

$$HV(f^{ref}, X) = \wedge\left(\bigcup_{X_n \in X}\left[f_1(X_n), f_1^{ref}\right] \times \cdots \times \left[f_m(X_n), f_m^{ref}\right]\right) \tag{24}$$

In Equation (24), f^{ref} represents the selected reference point and X represents the obtained approximation set, and the value obtained by summing its volumes is the HV index.

IGD is similar to Generational Distance (GD), but takes into account both diversity and convergence. IGD starts from the real PF and finds the point closest to it in the PF. This paper selects the optimal PF calculated by running all algorithms 10 times as the optimal PF. If the IGD index is smaller, it means that it is closer to the optimal PF, which means that the diversity and convergence of the algorithm are better. The IGD calculation method is shown in Equation (25).

$$IGD = \frac{\sum_{y \in PF^*} d(y, x)}{|PF^*|} \tag{25}$$

In Equation (25), PF^* represents the optimal Pareto front, x represents the obtained approximation set, and $d(y, x)$ represents each solution y in the true optimal PF. The solution x in PF that is closest to it is found, and its Euclidean distance is calculated.

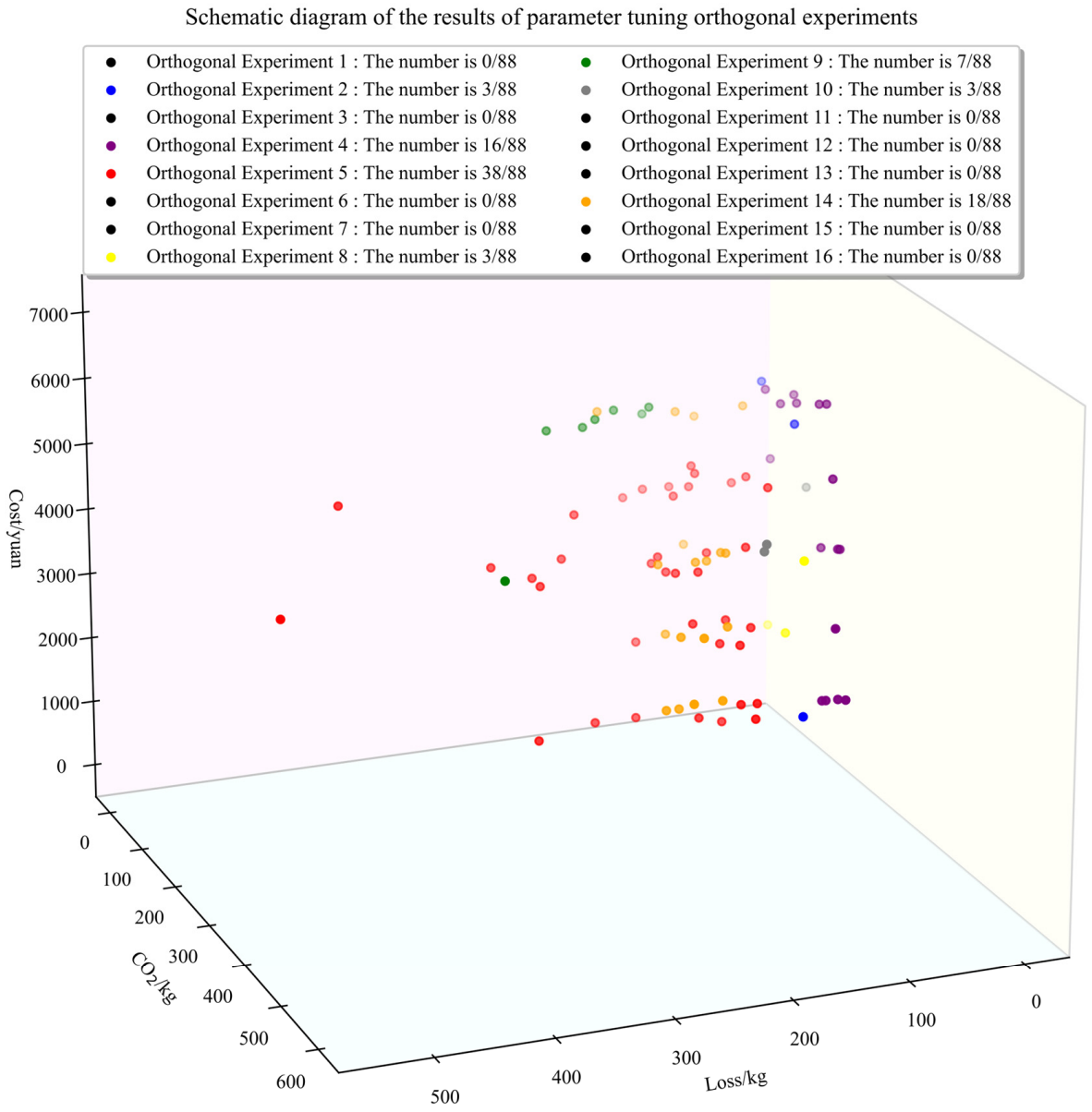

Figure 4. Parameter tuning result schematic.

The HV and IGD indicators for 40 groups of test examples across four sizes are shown in Table 4. The bolded values represent the optimal results for each calculation example. From Table 4, it can be seen that in medium- and large-scale calculation examples, the algorithm proposed in this article performs significantly better compared to the two hybrid algorithms. Additionally, the larger the scale, the more pronounced the improvement.

To visually compare the solution quality of MODRL-SIA and the comparison algorithm, the experimental results from LRPTW test examples at different scales are presented in Figure 5. The MODRL-SIA framework not only exhibits superior solution quality but also adeptly manages complex logistics challenges by dynamically adapting to varying operational conditions. This framework utilizes an advanced hybrid method that merges deep reinforcement learning with swarm intelligence algorithms, enabling efficient navigation through the complexities of logistics systems, including fluctuating demand, diverse

delivery requirements, and variable transportation conditions. By optimizing routes and resource allocations intelligently, the MODRL-SIA framework reduces costs and boosts operational efficiency, offering a robust solution to the multifaceted challenges of modern logistics. The demonstrated effectiveness of this algorithm in addressing complex logistics issues significantly enhances the credibility of our research findings and provides promising directions for future applications in the field.

Table 4. HV and IGD metrics of MODRL-SIA with comparison algorithms in different experiments.

Client Point Size	Experiment No.	MOPSO-GA		MOGA-VNS		MODRL-SIA	
		HV	IGD	HV	IGD	HV	IGD
Number of customer points is 30	Example 1	1.312	1.085	1.021	1.179	1.307	**1.084**
	Example 2	1.263	1.100	1.021	1.179	**1.362**	**1.060**
	Example 3	1.262	**1.096**	1.021	1.179	**1.297**	**1.096**
	Example 4	1.417	1.057	1.100	1.142	**1.449**	**1.038**
	Example 5	**1.394**	**1.068**	1.100	1.142	1.327	1.082
	Example 6	**1.425**	**1.061**	1.126	1.139	1.327	1.082
	Example 7	1.287	1.097	1.048	1.174	**1.424**	**1.049**
	Example 8	**1.381**	1.068	1.021	1.179	1.362	**1.060**
	Example 9	**1.438**	**1.058**	1.019	1.196	1.330	1.069
	Example 10	**1.488**	**1.043**	1.050	1.174	1.342	1.062
Number of customer points is 50	Example 11	1.282	1.159	0.856	1.296	**1.325**	**1.126**
	Example 12	1.037	1.219	0.769	1.335	**1.457**	**1.120**
	Example 13	1.133	1.182	0.808	1.298	**1.350**	**1.148**
	Example 14	**1.217**	1.196	1.013	1.209	1.208	**1.171**
	Example 15	1.078	1.191	0.808	1.298	**1.239**	**1.124**
	Example 16	1.010	1.233	1.013	1.209	**1.350**	**1.148**
	Example 17	1.287	1.142	0.805	1.299	**1.362**	**1.120**
	Example 18	1.133	1.176	0.849	1.297	**1.331**	**1.138**
	Example 19	1.147	1.179	0.851	1.297	**1.463**	**1.107**
	Example 20	1.069	1.207	0.809	1.336	**1.463**	**1.107**
Number of customer points is 80	Example 21	1.168	1.215	0.596	1.484	**1.448**	**1.080**
	Example 22	1.157	1.186	0.583	1.493	**1.481**	**1.064**
	Example 23	1.110	1.228	0.633	1.469	**1.327**	**1.143**
	Example 24	1.100	1.247	0.567	1.526	**1.503**	**1.073**
	Example 25	1.148	1.202	0.456	1.593	**1.503**	**1.073**
	Example 26	1.154	1.190	0.483	1.578	**1.503**	**1.073**
	Example 27	1.043	1.251	0.654	1.465	**1.503**	**1.073**
	Example 28	1.018	1.254	0.499	1.596	**1.327**	**1.143**
	Example 29	1.176	1.230	0.654	1.465	**1.183**	**1.192**
	Example 30	1.139	1.214	0.482	1.600	**1.502**	**1.060**
Number of customer points is 100	Example 31	1.031	1.257	0.391	1.688	**1.209**	**1.125**
	Example 32	1.054	1.254	0.308	1.774	**1.474**	**1.058**
	Example 33	1.137	1.219	0.436	1.664	**1.209**	**1.125**
	Example 34	1.07	1.209	0.480	1.585	**1.476**	**1.018**
	Example 35	1.076	1.227	0.422	1.684	**1.476**	**1.018**
	Example 36	1.042	1.230	0.452	1.640	**1.491**	**1.058**
	Example 37	1.027	1.250	0.492	1.612	**1.224**	**1.125**
	Example 38	1.058	1.227	0.492	1.612	**1.491**	**1.058**
	Example 39	1.168	1.192	0.492	1.612	**1.224**	**1.125**
	Example 40	1.052	1.234	0.492	1.612	**1.491**	**1.058**

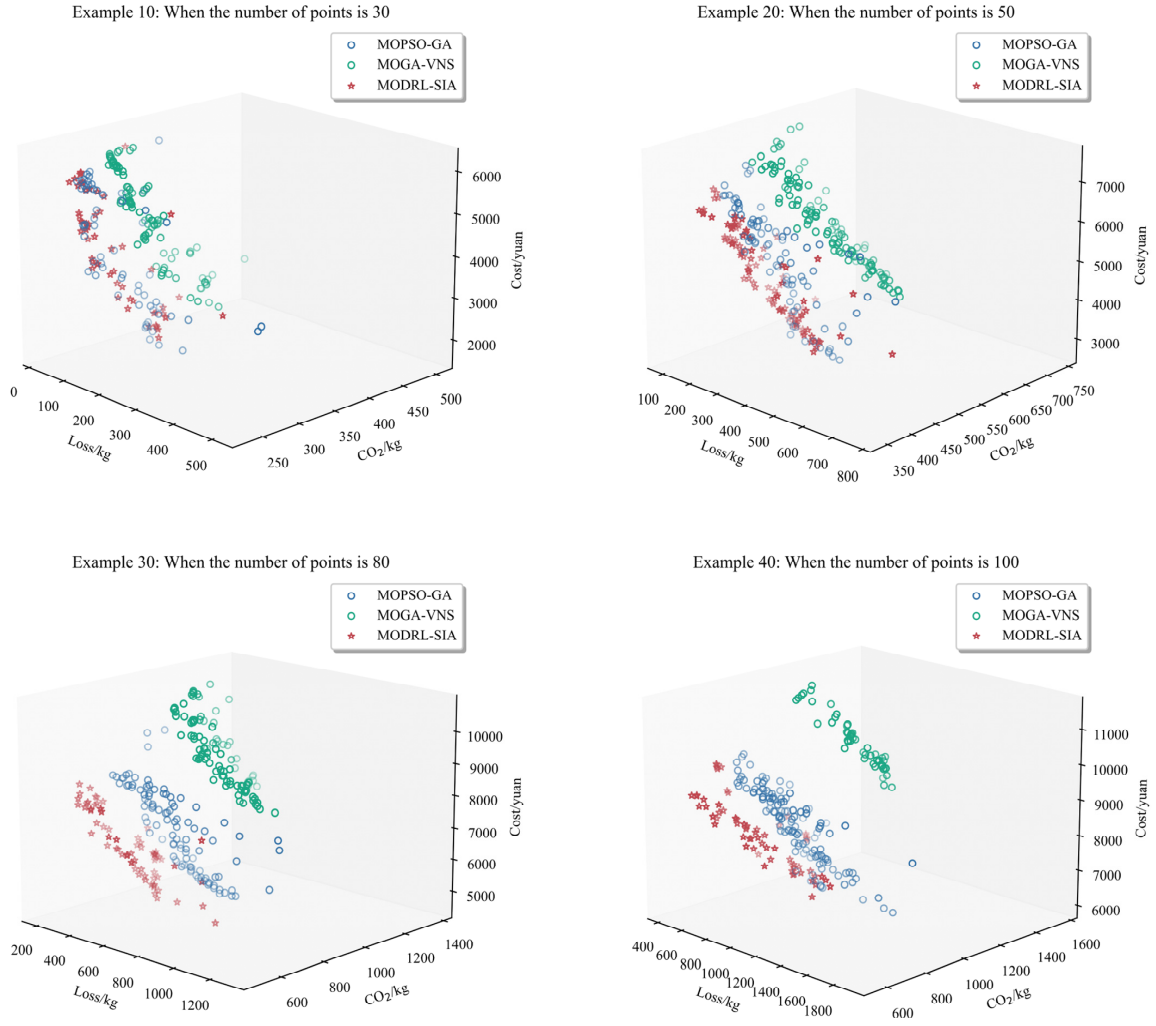

Figure 5. Comparison of Pareto frontiers at four scales.

6. Conclusions and Look Forward

In this paper, we present a novel DRL model tailored to solve the LRP for fresh produce. This model employs actors to generate and refine delivery solutions, with decreased training variance achieved through enhancements to the reward and punishment mechanisms. We developed mathematical models addressing multiple optimization objectives, including total operating costs, carbon emissions, and product losses. These models support the training of the DRL model using a fresh produce distribution simulator. Building on this foundation, we introduced the MODRL-SIA framework, integrating DRL with swarm intelligence algorithms to tackle the LRPTW. The framework offers superior accuracy, stability, and convergence speed compared to existing advanced hybrid algorithms, handling complex, large-scale challenges effectively.

Future research will progress in several directions: First, we aim to investigate advanced reinforcement learning techniques, such as attention mechanisms for multi-agent collaboration, to augment the MODRL-SIA framework's performance. Second, we intend to extend the range of objectives and constraints to encompass not only total distribution costs, carbon emissions, and product losses but also environmental impacts and traffic congestion. Furthermore, acknowledging the uncertainties in the agricultural product supply chain, including demand variability and climatic changes, upcoming studies will integrate uncertainty modeling and decision-making strategies. We also plan to leverage emerging technologies like the Internet of Things and artificial intelligence to craft innovative solutions for optimizing cold chain storage site selection and logistics routing decisions.

Additionally, given the prevalent risks in logistics systems, we will formulate efficient risk assessment models and product loss algorithms to manage risks and minimize economic losses effectively. Incorporating data analysis and machine learning techniques, future research will aim to elevate the intelligence and automation capabilities of logistics systems, providing more robust risk management and loss prevention strategies for logistics enterprises. These initiatives will propel the development of solutions for complex logistics routing and combinatorial optimization problems toward more comprehensive and efficient outcomes.

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

Funding: This study was financially supported by the Project of Gansu Natural Science Foundation (21JR7RA204 and 1506RJZA007) and the Gansu Province Higher Education Innovation Foundation (2022B-107 and 2019A-056).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Conflicts of Interest: The authors have no relevant financial or non-financial interests to disclose.

References

1. Office of the Third National Land Investigation Leading Group of The State Council; Ministry of Natural Resources; National Bureau of Statistics. Major data bulletin of the third National Land Survey. *People's Daily*, 27 August 2021; p. 017.
2. Statistics Bureau of the People's Republic of China. *China Statistical Yearbook*; China Statistics Press: Beijing, China, 2022.
3. Liu, Y.; Dang, Z.J.; Yao, J. Data Driven "Internet plus" Open Supply Chain System for Fresh Agricultural Products. In Proceedings of the 1st International Symposium on Management and Social Sciences (ISMSS), Wuhan, China, 13–14 April 2019; pp. 69–73.
4. Yan, B.; Chen, X.X.; Cai, C.Y.; Guan, S.Y. Supply chain coordination of fresh agricultural products based on consumer behavior. *Comput. Oper. Res.* **2020**, *123*, 105038. [CrossRef]
5. Fragomeli, R.; Annunziata, A.; Punzo, G. Promoting the Transition towards Agriculture 4.0: A Systematic Literature Review on Drivers and Barriers. *Sustainability* **2024**, *16*, 2425. [CrossRef]
6. Jiang, J.D.; Jiang, S.H.; Xu, G.Y.; Li, J. Research on Pricing Strategy and Profit-Distribution Mechanism of Green and Low-Carbon Agricultural Products' Traceability Supply Chain. *Sustainability* **2024**, *16*, 2087. [CrossRef]
7. Maghfiroh, M.F.N.; Yu, V.F.; Redi, A.; Abdallah, B.N. A Location Routing Problem with Time Windows Consideration: A Metaheuristics Approach. *Appl. Sci.* **2023**, *13*, 843. [CrossRef]
8. Yan, X.M.; Jin, Y.C.; Ke, X.H.; Hao, Z.F. Multi-task evolutionary optimization of multi-echelon location routing problems via a hierarchical fuzzy graph. *Complex Intell. Syst.* **2023**, *9*, 6845–6862. [CrossRef]
9. Hassanpour, S.T.; Ke, G.Y.; Zhao, J.H.; Tulett, D.M. Infectious waste management during a pandemic: A stochastic location-routing problem with chance-constrained time windows. *Comput. Ind. Eng.* **2023**, *177*, 109066. [CrossRef]
10. Sutrisno, H.; Yang, C.L. A two-echelon location routing problem with mobile satellites for last-mile delivery: Mathematical formulation and clustering-based heuristic method. *Ann. Oper. Res.* **2023**, *323*, 203–228. [CrossRef]
11. Tadaros, M.; Migdalas, A. Bi- and multi-objective location routing problems: Classification and literature review. *Oper. Res.* **2022**, *22*, 4641–4683. [CrossRef]
12. Han, B.; Shi, S.S.; Gao, H.T.; Hu, Y. A Sustainable Intermodal Location-Routing Optimization Approach: A Case Study of the Bohai Rim Region. *Sustainability* **2022**, *14*, 3987. [CrossRef]
13. Raeisi, D.; Ghoushchi, S.J. A robust fuzzy multi-objective location-routing problem for hazardous waste under uncertain conditions. *Appl. Intell.* **2022**, *52*, 13435–13455. [CrossRef]
14. Kordi, G.; Hasanzadeh-Moghimi, P.; Paydar, M.M.; Asadi-Gangraj, E. A multi-objective location-routing model for dental waste considering environmental factors. *Ann. Oper. Res.* **2023**, *328*, 755–792. [CrossRef] [PubMed]
15. Kallestad, J.; Hasibi, R.; Hemmati, A.; Soerensen, K. A general deep reinforcement learning hyperheuristic framework for solving combinatorial optimization problems. *Eur. J. Oper. Res.* **2023**, *309*, 446–468. [CrossRef]
16. Rotaeche, R.; Ballesteros, A.; Proenza, J. Speeding Task Allocation Search for Reconfigurations in Adaptive Distributed Embedded Systems Using Deep Reinforcement Learning. *Sensors* **2023**, *23*, 548. [CrossRef] [PubMed]

17. Fang, J.; Rao, Y.Q.; Luo, Q.; Xu, J.T. Solving One-Dimensional Cutting Stock Problems with the Deep Reinforcement Learning. *Mathematics* **2023**, *11*, 1028. [CrossRef]
18. Tu, C.F.; Bai, R.B.; Aickelin, U.; Zhang, Y.C.; Du, H.S. A deep reinforcement learning hyper-heuristic with feature fusion for online packing problems. *Expert Syst. Appl.* **2023**, *230*, 120568. [CrossRef]
19. Liu, H.; Zhao, F.Q.; Wang, L.; Xu, T.P.; Dong, C.X. Evolutionary Multitasking Memetic Algorithm for Distributed Hybrid Flow-Shop Scheduling Problem With Deterioration Effect. *IEEE Trans. Autom. Sci. Eng.* **2024**, 1–15. [CrossRef]
20. Fiege, N.; Kumm, M.; Zipf, P. Bit-Level Optimized Constant Multiplication Using Boolean Satisfiability. *IEEE Trans. Circuits Syst. I-Regul. Pap.* **2024**, *71*, 249–261. [CrossRef]
21. Jiang, S.Y.; Zou, J.; Yang, S.X.; Yao, X. Evolutionary Dynamic Multi-objective Optimisation: A Survey. *ACM Comput. Surv.* **2023**, *55*, 1–47. [CrossRef]
22. Khalid, A.M.; Hamza, H.M.; Mirjalili, S.; Hosny, K.M. MOCOVIDOA: A novel multi-objective coronavirus disease optimization algorithm for solving multi-objective optimization problems. *Neural Comput. Appl.* **2023**, *35*, 17319–17347. [CrossRef]
23. Shu, X.L.; Liu, Y.M.; Liu, J.; Yang, M.L.; Zhang, Q. Multi-objective particle swarm optimization with dynamic population size. *J. Comput. Des. Eng.* **2023**, *10*, 446–467. [CrossRef]
24. Pirouz, B.; Pirouz, B. Multi-Objective Models for Sparse Optimization in Linear Support Vector Machine Classification. *Mathematics* **2023**, *11*, 3721. [CrossRef]
25. Yang, Y.F.; Zhang, C.S. A Multi-Objective Carnivorous Plant Algorithm for Solving Constrained Multi-Objective Optimization Problems. *Biomimetics* **2023**, *8*, 136. [CrossRef]
26. Vignon, C.; Rabault, J.; Vinuesa, R. Recent advances in applying deep reinforcement learning for flow control: Perspectives and future directions. *Phys. Fluids* **2023**, *35*, 031301. [CrossRef]
27. Shen, S.G.; Wu, X.P.; Sun, P.J.; Zhou, H.P.; Wu, Z.D.; Yu, S. Optimal privacy preservation strategies with signaling Q-learning for edge-computing-based IoT resource grant systems. *Expert Syst. Appl.* **2023**, *225*, 120192. [CrossRef]
28. Fu, Q.M.; Li, Z.; Ding, Z.K.; Chen, J.P.; Luo, J.; Wang, Y.Z.; Lu, Y. ED-DQN: An event-driven deep reinforcement learning control method for multi-zone residential buildings. *Build. Environ.* **2023**, *242*, 110546. [CrossRef]
29. Zheng, Y.J. Water wave optimization: A new nature-inspired metaheuristic. *Comput. Oper. Res.* **2015**, *55*, 1–11. [CrossRef]
30. Huynh, D.C.; Ho, L.D.; Pham, H.M.; Dunnigan, M.W.; Barbalata, C. Water Wave Optimization Algorithm-Based Dynamic Optimal Dispatch Considering a Day-Ahead Load Forecasting in a Microgrid. *IEEE Access* **2024**, *12*, 48027–48043. [CrossRef]