Hindawi
Mathematical Problems in Engineering
Volume 2020, Article ID 9839634, 13 pages
https://doi.org/10.1155/2020/9839634

Research Article

Green Vehicle Routing and Scheduling Optimization of Ship Steel Distribution Center Based on Improved Intelligent Water Drop Algorithms

Gang Chen,[1] Xiaoyuan Wu,[1] Jinghua Li,[2] and Hui Guo[2]

[1]*Shanghai Waigaoqiao Shipbuilding Co., Ltd., Shanghai 200137, China*
[2]*College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China*

Correspondence should be addressed to Jinghua Li; likewalls@163.com and Hui Guo; hui0625@hrbeu.edu.cn

Received 24 May 2019; Revised 7 August 2019; Accepted 14 October 2019; Published 9 January 2020

Academic Editor: Georgios Dounias

The timeliness of the steel distribution center process contributes to the smooth progress of ship construction. However, carbon emissions from vehicles in the distribution process are a major source of pollution. Reasonable vehicle routing and scheduling can effectively reduce the carbon emissions of vehicles and ensure the timeliness of distribution. To solve this problem, a green vehicle routing and scheduling problem model with soft time windows was proposed in this study. An intelligent water drop algorithm was designed and improved and then compared with the genetic algorithm and the traditional intelligent water drop algorithm. The applicability of the improved intelligent water drop algorithm was demonstrated. Finally, this algorithm was applied to a specific example to demonstrate that the improved intelligent water drop algorithm effectively reduced the cost of such green vehicle problems, thus reducing the carbon emissions of vehicles during the distribution process and achieving reductions in environmental pollution. Ultimately, this algorithm facilitates the achievement of green shipbuilding.

1. Introduction

The shipbuilding industry is an intensive manufacturing industry that integrates labour, capital, and technology. The shipbuilding industry requires long working hours and has a high demand for resources and labour. This industry also exhibits a strong correlation with technology level. While providing tremendous wealth for human beings, the shipbuilding industry is also constantly producing pollutants, which have a serious negative impact on the environment. There are also several problems in the ship manufacturing process, such as waste of resources caused by the improper selection and utilization of materials and pollution of air, soil, and water. Therefore, it is necessary to promote green shipbuilding technology. Green shipbuilding is the necessary direction for shipbuilding industry reform.

The goal of green ship manufacturing is to minimize the amount of waste and harmful emissions produced during the life cycle of ship design, manufacture, service, and decommissioning, with the goal being to reduce pollution to air, water, and land. Green ship manufacturing considers the entire process of product demand, design, manufacture, and operation, while simultaneously considering the products, environment, resources, and human beings required and the environmental factors and resource utilisation efficiency of the process. Thus, green ship manufacturing strives to minimize the impacts on the environment and maximize the utilisation rate of resources, thereby providing economic and social benefits [1].

According to relevant records, logistical costs account for a large proportion of manufacturing industry costs in China; logistical costs can account up to 30%–40% of an enterprise's production costs. The shipbuilding industry is a large manufacturing industry, and the types and quantities of materials purchased have reached an astonishing level, which has led to higher logistical costs. Steel is a necessary product of the shipbuilding industry and is in enormous demand. Transportation costs are the logistical costs of steel procurement; other costs include handling, warehousing,

sorting, and distribution. In today's steel market, fluctuation in steel prices and the punctuality of distribution restrict the necessary reform and upgrading of China's shipbuilding industry. Establishment of a shipping steel processing and distribution center will greatly reduce the costs of shipyard inventory and logistics and improve the supply chain environment of marine steel.

Steel distribution is the main service of distribution centers; however, distribution services are also the main producers of carbon emissions. According to the TERM 2011 report issued by the European Environment Agency, in 2009, transport (including international shipping) accounted for 24% of the total greenhouse gas emissions in 27 European Union countries [2], and road transport accounted for 17% of the total greenhouse gas emissions [3]. Therefore, it is crucial to study carbon emissions in the steel distribution process [4, 5], as this process plays an important role in the development of green shipbuilding.

The vehicle routing problem (VRP) is a classical combinatorial optimisation problem that is common in many fields and applications. It was first proposed by Dantzig and Ramser in 1959 [6]. The goal is to minimise the total cost or transportation routes. Following an in-depth study of various practical problems, a large number of variations have been generated based on the traditional VRP problem; these include the capacity-constrained VRP [7], the VRP with time window (VRPTW) [8], the simultaneous delivery VRP [9], the stochastic VRP (SVRP) [10], and the stochastic demand VRP [11, 12]. At the same time, a variety of algorithms for these different types of problems have emerged. The most common solutions are the genetic algorithm [13, 14], the ant colony algorithm [15], the tabu search algorithm [16, 17], the two-stage method [18], and other heuristic algorithms. Many researchers have achieved significant improvements based on traditional heuristic algorithms [19–21].

Most of these traditional VRPs aim to minimise the fixed cost of vehicles, the cost of workers, or the number of vehicles used, and the driving route. The green vehicle routing and scheduling problem (GVRSP) was first proposed by Xiao and Konak in 2015 [22], and a more comprehensive version was published in 2016 [23]. However, the authors only provided a general situation, and this cannot meet the logistical distribution requirements of large-scale engineering projects typically represented by ship steel distribution centers. Zhang et al. [24] proposed a VRP that considered fuel consumption and carbon emissions. Specifically, the fuel cost, carbon emission cost, and vehicle use cost were incorporated into the traditional VRP problem, and a low-carbon path problem model was established. Jin and Jiang-Hua [25] studied a VRP with time windows that considered carbon emissions and speed optimisation. A speed-based carbon emission calculation method was introduced, and a mixed integer programming model was established, in which fuel, carbon emissions, and travel time costs were minimised.

Ship construction must be undertaken in strict accordance with production plans; otherwise, serious project delays and huge cost losses will occur. Ship steel distribution time requirements are very strict; thus, it is necessary to allocate greater penalties for early or late arrival to urge distribution centers to distribute on time, ensuring the smooth construction of ships. Therefore, given the insufficiency of the abovementioned research, combined with the actual demands of shipyards and steel distribution centers in shipbuilding enterprises, the present study fully considers distribution costs and carbon emissions to minimise carbon emissions and comprehensive costs. Thus, we establish a mixed integer linear programming (MILP) model for the green vehicle routing and scheduling problem with soft time window (GVRSP-STW).

The intelligent water drop (IWD) algorithm was first proposed by Shah-Hosseini [26] in 2007 to simulate the process by which water droplets in nature overcome obstacles and find a simple path to the ocean. In the process of optimisation, water droplets find the optimal path in the river by simulating the flow process of the river, changing their own velocity, carrying small amounts of soil, and constantly renewing the amount of soil in the river channel. The IWD algorithm has been applied in many natural science and engineering fields and has demonstrated significant advantages and potential. The IWD algorithm has also been applied to many combinatorial optimisation problems, including TSP [27], multiple knapsack [28], vehicle routing [29], and job shop scheduling [28, 30, 31], with satisfactory results. Kamkar et al. [32] used intelligent water droplet algorithm to solve the VRP problem in 2010. Fourteen benchmark VRP problems were experimented. The experimental results were compared with several other metaheuristic algorithms (simulated annealing algorithm, tabu search algorithm, and ant colony algorithm). The results show that the intelligent water droplet algorithm can converge to the optimal solution quickly and get better results. Therefore, in view of the problems raised in this paper, the intelligent water droplet algorithm is chosen to solve them.

The above research shows that the IWD algorithm is suitable for solving the VRP problem, providing greater efficiency and better results than other heuristic algorithms. However, to date, there is a scarcity of studies on solving VRP problems using this algorithm, and there is even less research applying this algorithm to the green vehicle routing optimisation problem.

In the current study, an improved IWD (IIWD) algorithm was designed to solve the green vehicle routing optimisation problem. A mutation operator was added to the traditional IWD algorithm to prevent the algorithm from falling into the local optimal solution; faster convergence is also achieved in solving the optimal solution [33]. The correctness and effectiveness of the model and the algorithm are demonstrated in this study. Finally, the IIWD algorithm was applied to the GVRSP-STW for the first time.

The remainder of the paper is organised as follows. Section 2 reviews the related literature. Section 3 describes the problem and proposes a mixed integer programming formulation. Section 4 gives details of the IIWD algorithm. Computational results for numerous instances are reported in Section 5. Conclusions and future research directions are provided in Section 6.

2. Problem Description and Formulation

2.1. Problem Description. The VRP between steel distribution centers and shipyards is shown in Figure 1 [34] and can be described as follows: a steel distribution center has a certain number of different types of vehicles, and steel is transported to several designated shipyards; the quantity of steel to be delivered to each shipyard is determined. Distribution centers assign vehicles to transport steel from distribution centers to shipyards according to the order requirements of different shipyards. Distribution centers need to comprehensively consider and make rational decisions on distribution vehicles, the distribution sequence, and the distribution scheme. Under the constraints of the maximum loading capacity of the vehicles and the need to satisfy the shipyard's delivery demand, the comprehensive costs, including the depreciation cost, manpower cost, and carbon emissions, are considered to achieve the lowest possible distribution costs.

The GVRSP-STW problem studied in the present study can be defined as follows. The known steel distribution center has a team of vehicles of the same type. The vehicles have a load capacity, fixed cost, carbon emissions, and other parameters. The distribution center serves N shipyards. The coordinates of each shipyard are known, and the demand is known. The distribution vehicles load steel from the distribution center and deliver goods to N shipyards. One vehicle must complete the delivery task for each shipyard. Each vehicle must start from the distribution center and return to the distribution center when the goods are delivered. The cost of each distribution vehicle consists of three parts: the fixed use cost of the vehicle, the cost of fuel and carbon emissions, and the driver's salary. The optimisation objective of this problem is to rationally allocate vehicles and arrange the vehicle distribution routes to minimise the total cost, taking into account both vehicle operating costs (including fixed use costs and driver's wages) and fuel and carbon emission costs.

2.2. Assumptions. Before further study, in order to facilitate the study of the problem and the establishment and solution of the model, the following assumptions are made:

(1) There is only one distribution center, and the location coordinates of the distribution center and the shipyards are known.

(2) The capacity of each vehicle can meet the needs of at least one shipyard at the same time, and the vehicles in the distribution center can meet the distribution needs of all shipyards.

(3) The load of each vehicle during transportation shall not exceed its maximum load.

(4) Vehicles in distribution centers are of the same type, and the maximum load is known.

(5) Each vehicle leaves the distribution center and returns to the distribution center after serving all shipyards. Each shipyard is allowed to visit only once.

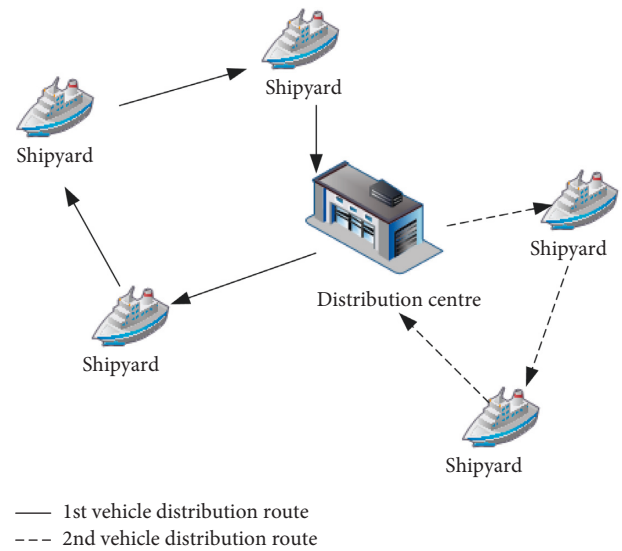

Figure 1: Schematic diagram of the vehicle routing problem between the steel distribution center and the shipyard.

(6) Vehicles can be loaded and unloaded immediately after arrival at the shipyard without waiting.

(7) The time for shipyards to allow distribution centers to provide services is fixed.

(8) The time spent in vehicle service is known, and the vehicle starts immediately after the shipyard completes the loading and unloading task.

2.3. Mathematical Model. Define a digraph as $G = (N, A)$, $N = \{1, 2, \ldots, n\}$ is a set of nodes, node 1 represents the distribution center, and $A = \{(i, j) i \neq j; i, j \in N\}$ represents the set of arcs between two nodes. $N_0 = \{2, 3, \ldots, n\}$ represents the shipyard node. There are $K = \{1, 2, \ldots, m\}$ vehicles of the same type in the yard, and the maximum capacity is Q. Each arc (i, j) represents the shortest distance between node i and node j, and the distance between arcs (i, j) is represented by d_{ij}. Each shipyard has two requirements: q_i reflects the weight of steel to be delivered to shipyard i by the distribution center, and shipyard i requires the distribution center to distribute steel during $[a_i, b_i]$ period. The arrival time of vehicle k at shipyard i is recorded as t_{ik}, and the service time of vehicle k at shipyard i is recorded as t_{sik}.

Therefore, the fuel and carbon emissions of vehicle k travelling between the arc ends (i, j) can be calculated according to the following formula [35]:

$$f_c = C_e d_{ij} \left[\rho_0 + \alpha Z_i^k\right], \tag{1}$$

where C_e is the cost of fuel and the carbon emissions per unit, d_{ij} is the distance of arc section (i, j), ρ_0 is the fuel consumption per unit distance to no-load vehicles, α is the additional fuel consumption per unit distance of goods loaded per unit weight, and Z_i^k is the total amount of cargo that must be delivered to all the shipyards to be visited after vehicle k arrives at shipyard i to complete the loading and unloading task.

Based on the above description and assumptions, the GVRSP-STW problem is modelled as a MILP model. The parameters required in the MILP model are listed in Table 1.

Decision variables:

x_{ij}^k: when vehicle k serves shipyard i and then goes to shipyard j for loading and unloading service, $x_{ij}^k = 1$; otherwise, $x_{ij}^k = 0$.

In summary, the total cost considered in this model consists of four parts:

(1) Fixed use costs of vehicles:

$$F_1 = C_v \sum_{j \in N_0} x_{0j}. \tag{2}$$

(2) Manpower expenditure of vehicles:

$$F_2 = C_p \sum_{k \in K} \sum_{i \in N_0} \sum_{j \in N_0} d_{ij} x_{ij}^k. \tag{3}$$

(3) Fuel and carbon emission costs:

$$F_3 = C_e \sum_{k \in K} \sum_{i \in N_0} \sum_{j \in N_0} d_{ij} x_{ij}^k \left[\rho_0 + \alpha Z_i^k\right]. \tag{4}$$

(4) Penalty cost for vehicle failure to arrive on time:

$$\mathrm{PC} = \mathrm{EP} \times \max\left(a_i - t_i^k, 0\right) + \mathrm{LP} \times \max\left(t_i^k - b_i, 0\right). \tag{5}$$

Therefore, the objective function can be expressed as

$$F = F_1 + F_2 + F_3 + \mathrm{PC}. \tag{6}$$

The constraints are

$$\sum_{j \in N_0} x_{0j}^k = 1, \quad \forall k \in K, \tag{7}$$

$$\sum_{k \in K} x_{ij}^k = 1, \quad \forall i \in N, \forall j \in N, \tag{8}$$

$$\sum_{k \in K} \sum_{j \in N} x_{ij}^k = 1, \quad \forall i \in N_0, \tag{9}$$

$$Z_0^k = \sum_{i \in N} \sum_{j \in N_0} q_j x_{ij}^k, \quad \forall k \in K, \tag{10}$$

$$Z_i^k = \sum_{l \in N} \left(Z_l^k - q_i\right) x_{li}^k, \quad \forall k \in K, \tag{11}$$

$$Z_i^k \le Q, \quad \forall k \in K, \forall i \in N, \tag{12}$$

$$\sum_{i \in N} x_{ij}^k = \sum_{m \in N} x_{jm}^k, \quad \forall j \in N_0, \tag{13}$$

Table 1

F	The objective function includes fixed use cost, fuel and carbon emission cost, and driver's salary
N	Set of nodes (including distribution centers)
N_0	Shipyard assembly
K	Vehicle assembly
d_{ij}	Distance of arc segment (i, j)
q_i	It means the weight of steel that the distribution center needs to deliver to shipyard i
$[a_i, b_i]$	Delivery time window allowed by shipyard i
ρ_0	The amount of fuel consumed per unit distance for no-load vehicles
α	Additional fuel consumption per unit distance for vehicles carrying goods per unit weight
t_i^k	The time when vehicle k arrives at shipyard i
t_{si}	The loading and unloading service time of vehicle k at shipyard i
Travel_{ij}^k	The travel time of vehicle k from node i to node j
Z_i^k	The total amount of cargo to be delivered by the shipyard after the vehicle k arrives at the shipyard i to complete the loading and unloading task
C_v	The fixed use cost of vehicles
C_p	Labour remuneration paid to drivers per unit time
C_e	Cost per unit of fuel and carbon emissions
EP	Penalty coefficient for early arrival of vehicles
LP	Penalty coefficient for vehicle lateness
PC	Penalty cost for vehicle failure to arrive on time

$$t_j^k = \sum_{j \in N} \left(t_i^k + t_{si}^k + \mathrm{Travel}_{ij}^k\right) x_{ij}^k, \quad \forall k \in K, \tag{14}$$

$$a_j \le t_j^k \le b_j. \tag{15}$$

Formulas (7)–(9) guarantee that each shipyard can only be served once; formula (10) denotes the total amount of cargo on board to be distributed to all subsequent shipyards when the vehicle departs the distribution center; formula (11) denotes the total amount of cargo to be delivered to all subsequent shipyards to be visited after the arrival of the vehicle at the shipyard; formula (12) denotes that the total amount of cargo carried by each vehicle cannot exceed the maximum capacity of the vehicle; formula (13) guarantees that the same vehicle arrives and leaves each shipyard; formula (14) indicates the time that the vehicle arrives at the shipyard; and formula (15) ensures that the arrival time of vehicle k at the shipyard must be greater than or equal to the earliest time window required by the shipyard and that the time after arrival at the shipyard to completion of unloading must be less than or equal to the latest time window required by the shipyard.

3. Improved Intelligent Water Drop Algorithm

The IWD algorithm simulates the movement of water droplets in nature and finds the optimised path. Each water droplet simulates the movement of a car. Each droplet starts

from the distribution center, chooses the next accessible node from the unavailable list, and updates the weight of the goods carried by the vehicle. When the weight of the water droplet exceeds the vehicle load or all accessible nodes are visited, the water droplet returns to the distribution center; this constitutes a completed vehicle route. The next intelligent water droplet starts to run until all nodes are visited and completed iteratively. Traditional IWD algorithms tend to converge slowly and often fall into obtaining a local optimum. To avoid this situation, an IIWD algorithm was designed. Elite strategy and variable neighborhood search mechanisms were added to ensure that the algorithm jumps out of local optimum and avoids prematurely converging.

In the IIWD, a water drop starts from distribution center 1, travels through various demand points along different paths, and finally returns to the distribution center to form a complete flow path (water droplets in the path can pass through distribution center 1 many times, but each demand point can only be passed through once). The complete flow path of each water droplet corresponds to a solution of the problem. If the complete flow path of a water drop is $1 \longrightarrow 6 \longrightarrow 2 \longrightarrow 1 \longrightarrow 3 \longrightarrow 4 \longrightarrow 7 \longrightarrow 1 \longrightarrow 9 \longrightarrow 5 \longrightarrow 8 \longrightarrow 1$, the distribution path of three vehicles is $1 \longrightarrow 6 \longrightarrow 2 \longrightarrow 1$, $1 \longrightarrow 3 \longrightarrow 4 \longrightarrow 7 \longrightarrow 1$, and $1 \longrightarrow 9 \longrightarrow 5 \longrightarrow 8 \longrightarrow 1$.

The specific design of the IIWD algorithm was as follows:

Step 1. Parameter initialization: static variables include droplet number N, initial velocity Initvel, iteration maximum number Iter, droplet velocity update parameters a_v,b_v, c_v, soil quantity update parameters a_s,b_s, c_s, local and global soil renewal coefficients ρ_0 and ρ_{IWD}, initial state path soil amount Initsoil, location coordinates of the distribution center and shipyards, demand of the shipyards, time window interval, distribution service time, time penalty coefficient, and other static variables in the model. Traditional IWD algorithms give the same initial velocity and the same initial path soil amount to each water drop, but it is well known that the choice of initial solution will have a certain impact on the quality of the solution. Therefore, the initial velocity of each water drop and the amount of soil along the path are generated randomly in the IIWD to ensure diversity of the initial solution and to improve the quality of the solution.

Step 2. The initial point is the distribution center node: according to the delivery time window and delivery volume requirements of other nodes, the required nodes are selected from the list of visiting nodes (UnvisitNode) and are stored in the list of accessible nodes (FitVisit).

Step 3. Calculation of the accessibility probability p_i of each node in the list of accessible nodes (FitVisit) and choosing the next serviceable node by roulette can improve the global search ability and avoid falling into a local optimum:

$$p_i = \frac{f\left(\mathrm{Soil}_{i,j}\right)}{\sum\limits_{k \in \mathrm{FitVisit}} f\left(\mathrm{Soil}_{i,k}\right)},$$

$$f\left(\mathrm{Soil}_{i,j}\right) = \frac{1}{\varepsilon + g\left(\mathrm{Soil}_{i,j}\right)},$$

$$g\left(\mathrm{Soil}_{i,j}\right) = \begin{cases} \mathrm{Soil}_{i,j}; & \text{if } \min\left(\mathrm{Soil}_{i,k}\right) \geq 0, \\ \mathrm{Soil}_{i,j} - \min\left(\mathrm{Soil}_{i,k}\right); & \text{else.} \end{cases} \tag{16}$$

In the formula, $\mathrm{Soil}_{i,j}$ represents the amount of soil on the path from node i to node j, the constant ε is 0.01, which ensures that the denominator of function $f(\mathrm{Soil}_{i,j})$ is not zero, function $g(\mathrm{Soil}_{i,j})$ ensures that the amount of soil is positive, and $\min(\mathrm{Soil}_{i,k})$ is the minimum amount of soil on the path from the current node to all serviceable nodes.

Step 4. Update the dynamic parameters of the intelligent water droplets: update the flow velocity $\mathrm{Vel}_{\mathrm{IWD}}(t)$ of the intelligent water droplets from the current node to the next selected node according to the following formula:

$$\mathrm{Vel}_{\mathrm{IWD}}(t) = \mathrm{Vel}_{\mathrm{IWD}}(t-1) + \frac{a_v}{b_v + c_v \times \mathrm{Soil}_{i,j}^2}. \tag{17}$$

In the formula, a_v, b_v, and c_v are the update parameters of droplet velocity, and $\mathrm{Soil}_{i,j}$ is the path soil amount from the current node to the selected next node.

Further, calculate the time $T_{i,j}$ needed for intelligent water droplets to flow to the target point and the increment of soil carried by water droplets $\Delta\mathrm{Soil}_{i,j}$, as shown in the following equation:

$$T_{i,j} = \frac{d_{i,j}}{\max\left(\varepsilon_v, \mathrm{Vel}_{\mathrm{IWD}}(t)\right)},$$
$$\Delta\mathrm{Soil}_{i,j} = \frac{a_s}{b_s + c_s \times T_{i,j}^2}. \tag{18}$$

In the formula, $d_{i,j}$ is the distance from the current node to the next selected node, constant ε_v ensures that the denominator is not zero, and a_s, b_s, and c_s are the update parameters of soil increment $\Delta\mathrm{Soil}_{i,j}$.

Step 5. Update the amount $\mathrm{Soil}_{i,j}$ of soil on the path through which water droplets flow: in the process of water droplet flow, the soil content in the path changes dynamically with the flow of water droplets. When the soil content in one path is much higher than that in other paths, the algorithm will converge too quickly and a better solution cannot be obtained. Similarly, when the soil content in a certain path is less, the convergence speed of the algorithm will slow. To prevent slow/premature convergence of the algorithm, referring to the bounded soil renewal model proposed by Niu et al. [30], the present study sets maximum and minimum limits on the amount of soil in each path, thereby slowing

down convergence when the amount of soil is too large and improving the convergence speed when the amount of soil is small. The formulas for calculating the amount of soil flowing through the path and the amount of soil carried by the water droplet in the IIWD algorithm are as follows:

$$\overline{\Delta}\mathrm{Soil}_{i,j} = \begin{cases} \Delta\mathrm{Soil}_{i,j}^{\min}; & \text{if } \Delta\mathrm{Soil}_{i,j} < \Delta Soil_{i,j}^{\min}, \\ \Delta\mathrm{Soil}_{i,j}^{\max}; & \text{if } \Delta\mathrm{Soil}_{i,j} > \Delta Soil_{i,j}^{\max}, \\ \Delta\mathrm{Soil}_{i,j}; & \text{else}, \end{cases} \tag{19}$$

$$\mathrm{Soil}_{i,j} = (1-\rho_0)\mathrm{Soil}_{i,j} - \rho_0\overline{\Delta}\mathrm{Soil}_{i,j},$$

$$\mathrm{Soil}_{i,j}^{\mathrm{IWD}} = \mathrm{Soil}_{i,j}^{\mathrm{IWD}} + \overline{\Delta}\mathrm{Soil}_{i,j}.$$

In the formula, $\Delta\mathrm{Soil}_{i,j}^{\min}$ and $\Delta\mathrm{Soil}_{i,j}^{\max}$, respectively, are the minimum and maximum limits of the increments of soil carried by water droplets; $\mathrm{Soil}_{i,j}$ represents the amount of soil on the path from node i to node j; $\mathrm{Soil}_{i,j}^{\mathrm{IWD}}$ represents the amount of soil carried by the water droplet from node i to node j; and ρ_0 is the local renewal coefficient (0.9).

Step 6. If the selected target node is the distribution center, the distribution vehicle returns to the distribution center. Otherwise, the visited list and the unavailable list are updated, and the algorithm returns to step 2 until the unavailable list is empty, indicating that all required nodes have been served and each water droplet has formed a complete access path. By calculating the value of the objective function, the minimum feasible solution of the objective function value is determined by comparison, which is the optimal solution S^{IB} of this iteration.

Step 7. Global soil quantity renewal: the soil amount on the path corresponding to the optimal solution obtained by this iteration is updated:

$$\mathrm{Soil}_{i,j} = (1+\rho_{\mathrm{IWD}})\mathrm{Soil}_{i,j} - \frac{\rho_{\mathrm{IWD}}\mathrm{Soil}_k^{\mathrm{IWD}}}{n-1}. \tag{20}$$

In the formula, ρ_{IWD} is the global soil quantity renewal coefficient, $\mathrm{Soil}_k^{\mathrm{IWD}}$ is the soil quantity carried by the water droplet corresponding to the k-th iteration optimal solution, and n is the number of nodes on the path corresponding to the iteration optimal solution.

Step 8. Update the global optimal solution:

$$S^{\mathrm{TB}} = \begin{cases} S^{\mathrm{TB}}; & \text{if } f(S^{\mathrm{TB}}) < f(S^{\mathrm{IB}}), \\ S^{\mathrm{IB}}; & \text{else}. \end{cases} \tag{21}$$

Step 9. The global optimal solution obtained by the IIWD algorithm is further optimized using the variable neighborhood search algorithm. By resetting and inserting the location of the sorting point in the neighborhood of the optimal path, a new route order is obtained, and satisfaction of the constraint for the sorting point in the distribution time and vehicle load after insertion is checked. Finally, a new path satisfying the constraint is obtained. To a certain extent, the IIWD algorithm can jump out of the local optimum and accelerate the convergence speed of the algorithm.

Step 10. Update the iteration coefficient: return to step 2 until the iteration is completed and output the optimal solution.

A flowchart of the implementation of the algorithm is given in Figure 2 [34].

4. Computational Experiments

In this section, the performance of the proposed method is evaluated by 100 randomly generated instances, which include small-(the scale of the problem is 30 nodes), medium-(the scale of the problem is 50 nodes), and large-(the scale of the problem is 100 nodes) sized problems. All algorithms were coded in MATLAB R2014b. All computational experiments were conducted on a PC with a 3.60 GHz processor and 8.00 GB RAM under the Windows 10 operating system.

To demonstrate the effectiveness and superiority of the IIWD droplet algorithm proposed in the present study, the GA, traditional IWD algorithm, and IIWD algorithm were compared. The problem group R101–R112 is based on the Solomon VRPTW standard example [36, 37]; the penalty coefficients for early arrival and late arrival are added to the algorithm to satisfy the mathematical model proposed in the present study. The design of these coefficients encompasses several factors that affect the behaviour of routing and scheduling algorithms, including geographical data, the number of customers serviced by a vehicle, percent of time-constrained customers, and tightness and positioning of the time windows. The problems differ with respect to the width of the time windows. Some have very tight time windows, while others have time windows which are hardly constraining. Table 2 shows the definitions and numerical values of some of the parameters needed in the application of the IWD algorithm. The number of iterations was set at 50; the results are shown in Table 3.

The GA uses integer string coding; the order of integers represents the order in which the vehicle visits the shipyards. The roulette method is used to select the selection operator, and the greater the fitness, the greater the probability of crossover operation. The crossover probability and mutation probability in this study were 0.6 and 0.05, respectively.

The simulation results show that the IIWD algorithm has faster convergence speed and higher efficiency than the traditional IWD algorithm when solving the GVRSP-STW problem. This is because the improved algorithm can reset and insert some nodes from the better solution after iteration; this overcomes the problem of long search time of the traditional IWD algorithm and effectively avoids falling into a local optimum in the search process. As can be seen from Table 3, compared with the GA and the traditional IWD algorithm, the IIWD algorithm is superior in terms of the total cost, vehicle driving distance, and vehicle usage. The total cost and the number of vehicles used are the lowest under the IIWD; this will reduce carbon emissions, thereby achieving the goal of green shipbuilding.

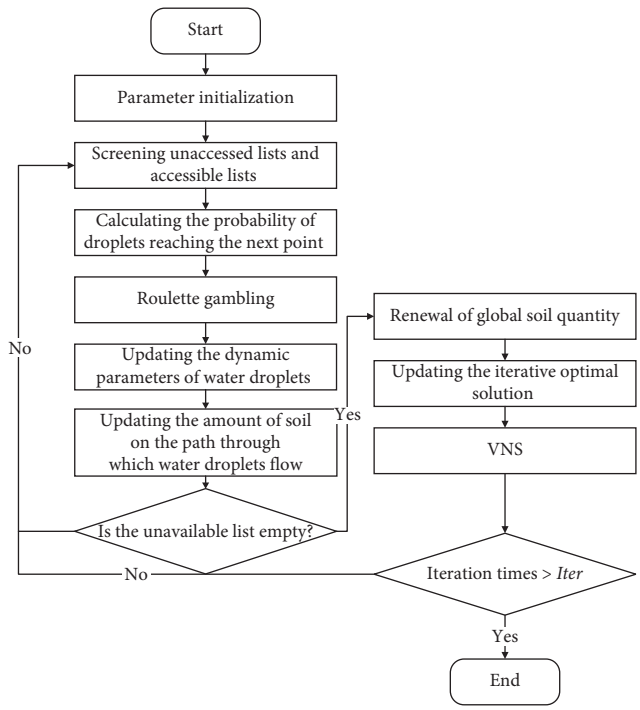

Figure 2: The flowchart of the implementation of the algorithm.

Table 2: Initialization parameters of IIWD.

Parameters	Meanings	Value
Q	Vehicle rated load	49t
v_k	Average vehicle speed	40 km/t
C_v	Fixed-use costs of vehicles	200/one
C_p	Unit transportation cost per vehicle	10/km
N_{IWD}	Number of drops	20
a_v	Droplet velocity updating parameters	1
b_v	Droplet velocity updating parameters	0.01
c_v	Droplet velocity updating parameters	1
ρ_0	Fuel consumption per unit distance for no-load vehicles	2
a_s	Renewal parameters of soil and water droplets	1000
b_s	Renewal parameters of soil and water droplets	0.01
c_s	Renewal parameters of soil and water droplets	1
Initsoil	Initial path amount of soil	1000
Initvel	Initial droplet velocity	100
ρ_0	Local soil renewal parameters	0.9
ρ_{IWD}	Global soil renewal parameters	0.8
Iter	Maximum number of iterations	100
α	Additional fuel consumption per unit distance for vehicles carrying goods per unit weight	0.8

As can be seen from Table 3, for the standard Solomon examples, the IIWD algorithm has obvious advantages over the GA and the traditional IWD algorithm. The IIWD algorithm outperforms the other two algorithms in terms of total driving distance and total cost. Therefore, the application of the IIWD algorithm in vehicle scheduling calculation can effectively reduce the cost, reduce the vehicle travel distance, further reduce carbon emissions, and greatly reduce vehicle air pollution in the distribution process, thus achieving the goal of green shipbuilding (Table 4 Figures 3–5).

Based on the above, a steel distribution center in Shanghai and 17 shipyards around it were used as an example. The locations of the distribution center and shipyards were determined with Baidu Maps, and the customer requirements and acceptable time windows are known. The IWD and the IIWD algorithm designed in the present study were each used to solve the problem. The

Table 3: Relevant data.

Problem group	GA				IDW				IIDW			
	Distance	Cost	C cost	CPU time	Distance	Cost	C cost	CPU time	Distance	Cost	C cost	CPU time
R101	3697.9	172030	35464	**3.01** s	3500.6	206140	45120	5.92 s	**3371.7**	**166230**	**33420**	3.26 s
R102	3555.9	186180	36150	**3.59** s	3535.3	216150	44311	6.35 s	**3483.6**	**174330**	**34551**	4.01 s
R103	3686.9	185860	36981	3.86 s	3644.8	203230	46102	4.29 s	**3319.1**	**177212**	**36806**	**3.53** s
R104	3650.5	181471	39760	**3.20** s	3685.1	209840	47230	5.13 s	**3420.6**	**167178**	**38083**	4.03 s
R105	3536.1	172480	38012	**3.62** s	3451.8	207270	45620	5.53 s	**3391.6**	**166847**	**36692**	3.73 s
R106	3377.1	160080	41320	**3.73** s	3556.6	208840	48158	4.36 s	**3241.3**	**147106**	**40281**	3.90 s
R107	3662.9	179020	33015	3.23 s	3414.9	196120	43012	5.02 s	**3319.4**	**167955**	**31304**	**3.01** s
R108	3506	177732	37108	4.01 s	3225.9	198400	45890	4.88 s	**3094.2**	**168048**	**35655**	**3.95** s
R109	3237.9	189310	36550	**3.92** s	3223.4	216650	46301	5.59 s	**3071.6**	**176759**	**36263**	4.25 s
R110	3351.4	194700	37163	**4.62** s	3392.4	187280	45285	6.08 s	**3131.3**	**177486**	**35445**	5.09 s
R111	3547	187390	40103	**4.44** s	3390.7	197750	48162	6.64 s	**3230.2**	**178267**	**38075**	4.99 s
R112	3749.3	184180	42560	**3.63** s	3583.8	217470	49032	5.20 s	**3479.7**	**177696**	**40022**	4.29 s

Bold type represents the best value in each group of data.

Table 4: Optimal routes.

Type	Best route
R101	1-83-49-50-93-94-97-17-87-15-46-47-37-84-100-101-25-13-1-39-58-20-79-27-90-85-81-30-28-2-54-12-91-21-35-4-1-29-59-99-95-82-71-7-14-23-75-36-57-69-80-32-9-8-53-1-41-40-74-3-10-66-33-65-48-52-43-22-1-18-34-78-76-6-51-60-92-45-77-11-89-1-61-64-73-1-98-86-5-42-38-88-44-16-31-19-63-70-55-1-96-1-62-24-68-26-56-72-67-1
R102	1-53-87-86-69-1-93-1-98-19-20-12-51-5-85-27-57-74-100-83-49-55-61-79-30-56-1-25-78-71-76-75-14-7-9-59-90-80-77-29-48-41-35-18-1-54-28-58-72-66-36-1-60-81-34-13-97-96-6-94-21-67-62-47-1-37-50-42-95-1-2-40-24-68-26-52-89-1-99-44-38-88-3-46-17-101-23-84-91-70-22-73-8-4-82-10-16-1-11-43-15-39-45-32-31-1-63-65-64-33-1-92-1
R103	1-59-38-24-68-40-13-16-1-83-49-11-19-61-1-56-91-79-1-4-23-95-57-73-1-32-42-96-99-72-64-58-94-34-82-66-52-92-1-51-70-2-21-60-55-12-65-67-84-100-93-17-43-29-7-62-46-47-9-1-26-5-3-75-41-54-6-85-90-77-39-1-36-81-76-86-98-71-33-22101-8-53-1-88-78-80-35-30-69-74-97-63-31-10-28-89-48-1-25-14-87-44-15-45-18-37-50-20-27-1
R104	1-47-89-33-7-95-49-83-23-2-31-97-28-91-71-1-76-3-75-41-70-52-40-68-56-21-88-54-29-19-96-26-1-85-77-12-50-37-48-63-10-57-35-82-34-59-22-55-25-1-58-16-5-90-64-65-20-92-15-9-46-87-39-44-43-6-24-42-53-72-66-80-4-1-8-45-1-13-78-30-86-93-98-74-17-61-11-62-81-27-36-100-1-32-69-38-14-101-99-60-84-51-1-79-73-1-67-18-94-1
R105	1-34-82-70-61-85-5-81-90-101-89-29-32-75-74-22-13-4-93-2-1-14-94-35-25-36-66-72-78-86-58-23-77-1-11-31-21-71-100-97-7-10-45-96-83-49-92-1-73-68-40-28-69-1-79-41-1-80-67-33-64-50-37-48-9-46-17-95-1
R106	1-28-30-58-44-39-17-60-27-49-71-52-43-85-9-56-73-47-1-96-97-59-55-21-12-50-37-48-93-88-100-19-83-46-10-7-1-64-6-54-1-78-25-82-33-8-18-87-99-94-61-20-65-67-22-45-95-1-31-16-86-76-57-40-69-35-4-1-90-101-15-98-38-5-80-84-36-53-79-51-89-70-68-81-1-92-2-14-26-13-32-63-11-72-66-91-1-29-75-74-41-77-1-34-3-62-23-24-42-1
R107	1-53-70-2-1-19-1-86-88-23-74-22-98-76-75-7-47-8-32-84-48-37-50-20-93-94-1-15-39-87-99-100-25-40-68-24-57-63-14-38-59-18-79-77-29-90-81-1-3-55-12-64-91-35-6-95-62-13-31-71-61-101-1-56-26-30-36-60-83-33-9-34-45-17-27-1-42-16-1-51-54-46-89-21-1-11-41-78-80-4-58-52-82-5-96-1-10-66-69-73-1-97-1-49-43-44-92-85-28-1-65-67-72-1
R108	1-79-4-58-78-49-64-89-9-50-65-5-40-68-24-42-96-31-1-26-62-86-38-88-7-8-48-21-91-70-76-74-90-34-13-23-93-1-43-15-39-87-100-97-84-14-16-44-47-54-55-6-56-25-1-57-83-53-28-27-71-73-1-61-60-19-12-63-1-98-95-20-29-59-32-1-11-41-33-67-30-69-81-17-45-92-77-80-85-18-1-52-22-75-101-3-94-37-46-10-36-66-72-99-51-82-35-2-1
R109	1-95-75-16-74-25-35-51-61-100-96-10-72-21-64-93-101-45-2-90-7-1-67-52-11-32-23-44-15-39-87-78-48-1-12-28-99-71-46-62-86-3-59-36-79-81-53-8-98-97-66-9-58-65-1-60-92-29-69-13-38-42-43-17-88-73-56-1-5-54-31-63-94-26-55-84-83-89-19-70-30-1-34-82-18-6-76-40-4-1-49-1-20-77-27-33-91-50-37-47-85-1-22-24-68-57-41-80-14-1
R110	1-84-46-22-83-95-77-82-70-24-76-73-64-56-8-75-74-54-1-38-101-87-97-2-79-80-59-72-66-35-23-3-96-15-36-91-1-30-71-29-13-67-25-94-100-51-78-12-11-93-1-28-50-47-85-53-49-48-37-33-63-32-55-90-81-5-57-19-65-26-58-61-1-40-68-4-41-21-31-92-39-89-1-14-62-86-60-99-18-98-42-16-43-45-1-10-34-27-9-6-44-88-20-1-7-1-69-17-52-1
R111	1-13-86-18-100-81-61-6-98-1-85-36-72-67-88-94-71-77-4-65-91-12-33-52-82-1-55-79-35-66-53-15-84-23-39-38-93-49-92-76-58-78-80-1-11-64-9-46-70-29-48-83-42-24-68-40-57-99-74-73-43-45-87-19-1-26-56-28-90-8-21-75-1-14-10-7-59-41-97-69-30-51-2-32-63-22-1-17-60-95-50-20-27-44-25-3-96-34-1-54-31-16-1-5-101-62-47-37-89-1
R112	1-26-40-73-83-68-91-62-23-57-2-61-16-41-56-55-12-1-20-50-37-8-81-27-69-75-42-44-4-97-7-51-34-82-54-70-1-84-14-38-28-65-63-48-11-22-24-1-53-46-90-25-94-60-33-64-49-78-30-79-93-1-66-1-98-17-87-101-88-74-32-92-71-85-6-95-5-59-89-1-21-31-96-76-43-45-86-52-77-1-9-99-19-13-3-80-35-72-1-47-18-39-15-58-100-1-29-67-36-10-1

distribution center uses six-axle trucks for its distribution service. According to the national truck load standard, the rated load of six-axle trucks is 50 tonnes. Specific data are shown in Table 5 (serial number 1 indicates the distribution center point), and the initialization parameters are the same as in Table 2.

The example was programmed using MATLAB2016 software, and analysis was performed on a PC with 8G RAM

Figure 3: Shortest distance contrast chart.

Figure 4: Minimum carbon cost chart.

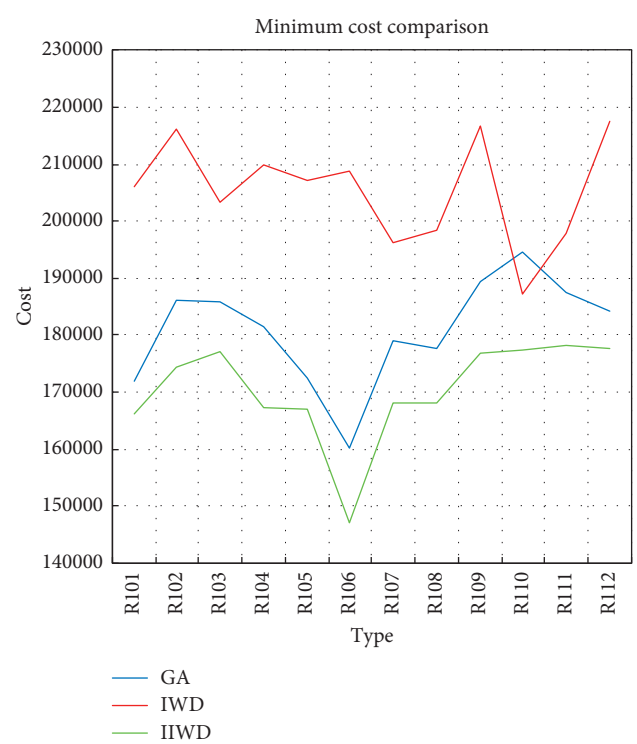

Figure 5: Optimal total cost contrast chart.

and Windows 8.1 operating system. The specific running iteration process is shown in Figure 6.

As can be seen from Figure 6, the IIWD algorithm obtains the optimal solution after 20 iterations. Currently, there are three vehicles for distribution. The specific allocation scheme and vehicle routing are shown in Tables 6 and 7. It can be seen that the total cost of the IIWD algorithm is 2041.6 less than that of the traditional IWD algorithm; this is an approximate cost saving of

Table 5: Relevant data of the distribution center and the shipyard.

Distribution point	X coordinates	Y coordinates	Requirement	Service time	Time window	Early arrival penalty coefficient	Late arrival penalty coefficient
1	18.7	15.29	0	0	(0, 40)	0	0
2	16.47	8.45	6	1.8	(5, 20)	5	15
3	20.07	10.14	5	1	(4, 15)	6	30
4	19.39	13.37	11	2.3	(1, 20)	10	10
5	25.27	14.24	6	1.8	(2, 20)	5	16
6	22	10.04	3	1.2	(5, 15)	8	18
7	25.47	17.02	8	2.4	(2, 18)	11	32
8	15.79	15.1	5	1.5	(1, 24)	8	21
9	16.6	12.38	6	1.8	(3, 27)	9	27
10	14.05	18.12	4	1.2	(1, 20)	12	15
11	17.53	17.38	5	1.5	(2, 16)	6	18
12	23.52	13.45	7	2.1	(2, 20)	7	22
13	19.41	18.13	6	1.8	(2, 10)	8	25
14	22.11	12.51	10	2	(3, 25)	7	31
15	11.25	11.04	9	2.7	(2, 28)	11	10
16	14.17	9.76	4	1.3	(3, 24)	5	10
17	24	19.89	7	1.2	(2, 20)	4	15
18	12.21	14.5	8	1.5	(1, 23)	6	18

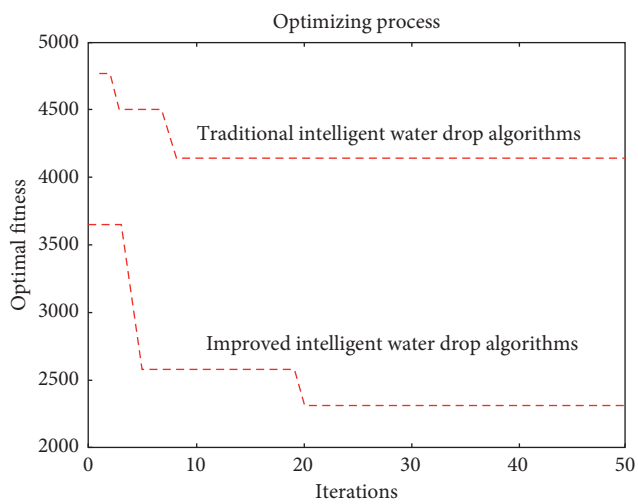

Figure 6: Iterative contrast diagram of algorithms.

Table 6: Vehicle optimal routing and total cost (traditional intelligent water drop algorithms).

Vehicle number	Route	Total cost	Total distance	Total C cost
1	15-18-8-9			
2	14-5-7			
3	12-17	4348	117.17	821.36
4	6-3-2-16			
5	10			
6	11-13-4			

Table 7: Optimal vehicle routing and total cost (improved intelligent water drop algorithms).

Vehicle number	Route	Total cost	Total distance	Total C cost
1	4-3-6-14-12-5-17			
2	13-11-7	2306.4	78.098	463.99
3	9-8-2-16-15-18-10			

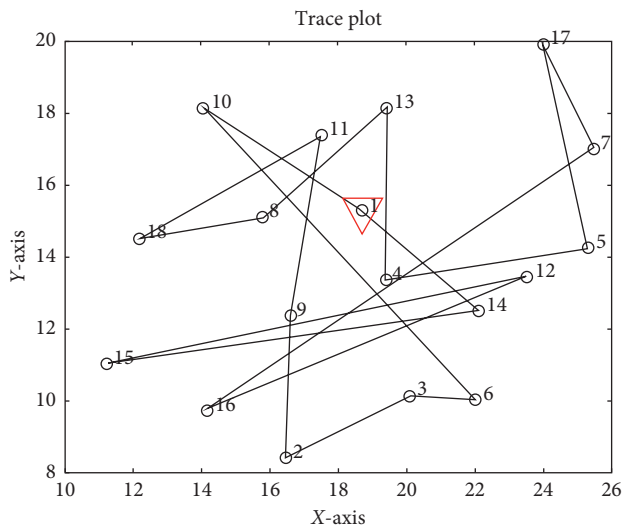

Figure 7: Road map of traditional intelligent water drop algorithms.

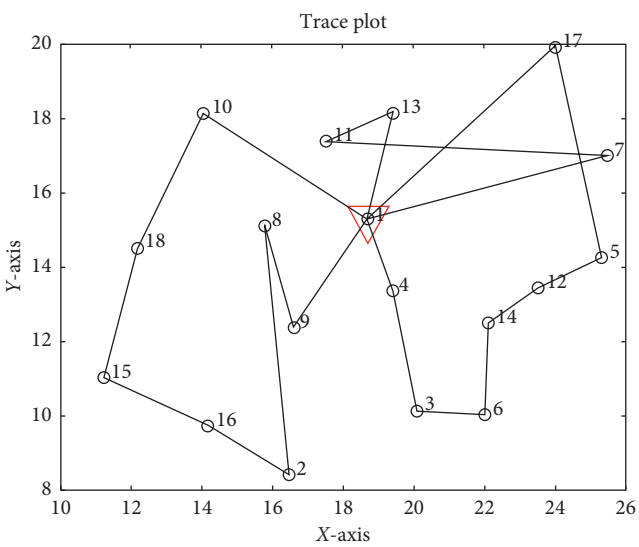

Figure 8: Road map for improved intelligent water drop algorithms.

20%. In terms of the total distance, the IIWD algorithm runs at a distance 11.353 less than the traditional IWD algorithm. In terms of the number of vehicles used, the number of vehicles used in the IIWD algorithm is half that of the traditional IWD algorithm. In conclusion, when solving the GVRSP-STW problem, the IIWD algorithm exhibits faster convergence speed and higher efficiency than the traditional IWD algorithm (Figures 7 and 8).

5. Conclusions

Ship steel distribution is a very important link during ship construction. Timely and effective distribution can not only save a lot of time but also ensure the smooth progress of the ship construction plan. Additionally, the effective control of carbon emissions during distribution can reduce environmental pollution, increase the impact of green shipbuilding, and improve the competitiveness of shipyards. To solve the vehicle routing and dispatching problem of a ship steel distribution center under the green shipbuilding mode, based on the conventional VRP model, the amount of carbon dioxide emitted by vehicles was set as one of the optimisation objectives. Due to the high requirements for ship construction timeliness, the penalty costs of early arrival or late arrival were also proposed based on a soft time window. A mathematical model of total cost optimisation was established in this study, the aim of which was to optimise vehicle depreciation and humanpower costs and carbon emissions. A GVRSP optimisation method based on our IIWD was proposed. In comparison to the traditional IWD algorithm and GA, we demonstrated that the GVRSP optimisation method based on the IIWD proposed in this study greatly reduced the total cost of vehicle distribution and the total vehicle running distance. This reduced the carbon emissions involved in vehicle distribution while ensuring the progress of ship construction. This meets the requirements of green environmental protection, thus achieving green shipbuilding. The GVRSP optimisation method based on the IIWD proposed in the present study has certain practical value and theoretical significance. In practice, there cannot be only one distribution center. In order to simplify the calculation, this paper assumes that there is only one distribution center. In the follow-up study, the problem of multiple distribution centers will be studied. In practice, the distribution center cannot have only one type of vehicle. In order to simplify the model and facilitate calculation, this paper assumes that there is only one type of vehicle in the distribution center. In the follow-up study, the multitype vehicle problem can be studied. Vehicles may not be loaded and unloaded immediately after arrival at the shipyard, and there may be delays. This paper assumes that there is no waiting time, and the factors of service waiting time will be taken into account in subsequent studies. Similarly, follow-up studies should take into account such factors that the service time of each shipyard may vary. In our next study, the problem model will be closer to the real distribution process to make the results more applicable. In follow-up research, we plan to consider the multispecifications and dynamic time window of multidistribution centers in the mathematical model, and we plan to design a corresponding algorithm to solve the model to further improve the practicability of the research.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Gang Chen and Xiaoyuan Wu contributed equally to this work.

Acknowledgments

This research was funded by the National Natural Science Foundation of China (no. 51679059), "Research on Intelligent Manufacturing Solution and Key Technologies for Upper Module of Offshore Oil and Gas Production Platform" (no. 2018473), "Research on Key Common Technologies towards Smart Manufacturing in Shipbuilding Industry" (no. 2016543), and "Research Project of High Tech Ship supported by the Ministry of Industry and Information Technology of the People's Republic of China" (no. 2016545).

References

[1] J. Liu, "Green ship and green shipbuilding technological innovation and development," in *Proceedings of the Sixteenth Annual Conference of China Association of Science and Technology—Papers Collection of 8 Forum on Green Shipbuilding and Safe Shipping*, p. 7, China association of science and technology, Yunnan Provincial People's Government: Academic Department of China Association of Science and Technology, 2014.

[2] A. Sanchez Vicente, "A closer look at urban transport. Term 2013: transport indicators tracking progress towards environmental targets in Europe," Eea Report, European Environment Agency, Copenhagen, Denmark, 2013.

[3] Li Li and J. Xing, "Research on low carbon logistics development in China under low carbon economic trend," *Business Age*, vol. 31, pp. 24-25, 2011.

[4] X. Guo and X. You, "Low carbon distribution center's location decision method under carbon emissions constraint condition," *Journal of Applied Sciences*, vol. 13, no. 11, pp. 1988–1991, 2013.

[5] T. C. Kuo, G. Y.-H. Chen, M. L. Wang, and M. W. Ho, "Carbon footprint inventory route planning and selection of hot spot suppliers," *International Journal of Production Economics*, vol. 150, no. 150, pp. 125–139, 2014.

[6] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.

[7] J. Lysgaard, A. N. Letchford, and R. W. Eglese, "A new branch-and-cut algorithm for the capacitated vehicle routing problem," *Mathematical Programming*, vol. 100, no. 2, pp. 423–445, 2004.

[8] O. Braysy and M. Gendreau, "Vehicle routing problem with time windows, part i: route construction and local search algorithms," *Transportation Science*, vol. 39, no. 1, pp. 104–118, 2005.

[9] H. Min, "The multiple vehicle routing problem with simultaneous delivery and pick-up points," *Transportation Research Part A: General*, vol. 23, no. 5, pp. 377–386, 1989.

[10] W. H. Yang, K. Mathur, and R. H. Ballou, "Stochastic vehicle routing problem with restocking," *Transportation Science*, vol. 34, no. 1, pp. 99–112, 2000.

[11] M. Gendreau, G. Laporte, and R. Séguin, "Stochastic vehicle routing," *European Journal of Operational Research*, vol. 88, no. 1, pp. 3–12, 1996.

[12] D. Teodorovic and G. Pavkovic, "A simulated annealing technique approach to the vehicle routing problem in the case of stochastic demand," *Transportation Planning and Technology*, vol. 16, no. 4, pp. 261–273, 1992.

[13] B. M. Baker and M. A. Ayechew, "A genetic algorithm for the vehicle routing problem," *Computers & Operations Research*, vol. 30, no. 5, pp. 787–800, 2003.

[14] I. Sbai, O. Limem, and S. Krichen, "An adaptive genetic algorithm for the capacitated vehicle routing problem with time windows and two-dimensional loading constraints," in *Proceedings of the ACS International Conference on Computer Systems & Applications*, IEEE, Hammamet, Tunisia, October 2018.

[15] J. E. Bell and P. R. Mcmullen, "Ant colony optimization techniques for the vehicle routing problem," *Advanced Engineering Informatics*, vol. 18, no. 1, pp. 41–48, 2004.

[16] J. Renaud, F. F. G. Laporte, and F. F. Boctor, "A tabu search heuristic for the multi-depot vehicle routing problem," *Computers & Operations Research*, vol. 23, no. 3, pp. 229–235, 1996.

[17] Y. Xia and Z. Fu, "Improved tabu search algorithm for the open vehicle routing problem with soft time windows and satisfaction rate," *Cluster Computing*, vol. 22, no. S4, pp. 8725–8733, 2018.

[18] J. Zhang, Y. Zhao, D. Peng et al., "A hybrid quantum-inspired evolutionary algorithm for open vehicle routing problem," in *Proceedings of the 2009 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, IEEE, Singapore, July 2009.

[19] N. M. E. Normasari, V. F. Yu, C. Bachtiyar, and Sukoyo, "A simulated annealing heuristic for the capacitated green vehicle routing problem," *Mathematical Problems in Engineering*, vol. 2019, no. 2, Article ID 2358258, 18 pages, 2019.

[20] M. Nazari, A. Oroojlooy, L. V. Snyder et al., *Reinforcement Learning for Solving the Vehicle Routing Problem*, Cornell University, Ithaca, NY, USA, 2018.

[21] A. Subramanyam, C. E. A. Wang, and C. E. Gounaris, "A scenario decomposition algorithm for strategic time window assignment vehicle routing problems," *Transportation Research Part B: Methodological*, vol. 117, pp. 296–317, 2018.

[22] Y. Xiao and A. Konak, "A simulating annealing algorithm to solve the green vehicle routing & scheduling problem with hierarchical objectives and weighted tardiness," *Applied Soft Computing*, vol. 34, pp. 372–388, 2015.

[23] Y. Xiao and A. Konak, "The heterogeneous green vehicle routing and scheduling problem with time-varying traffic congestion," *Transportation Research Part E: Logistics and Transportation Review*, vol. 88, pp. 146–166, 2016.

[24] J. Zhang, Y. Zhao, W. Xue, and J. Li, "Vehicle routing problem with fuel consumption and carbon emission," *International Journal of Production Economics*, vol. 170, pp. 234–242, 2015.

[25] L. I. Jin and Z. Jiang-Hua, "Vehicle routing problem with time windows based on carbon emissions and speed optimization," *Systems Engineering-Theory & Practice*, vol. 34, no. 12, pp. 3063–3072, 2014.

[26] H. Shah-Hosseini, "Problem solving by intelligent water drops," in *Proceedings of the IEEE Congress on Evolutionary Computation*, IEEE, Singapore, September 2007.

[27] B. O. Alijla, L.-P. Wong, C. P. Lim, A. T. Khader, and M. A. Al-Betar, "A modified intelligent water drops algorithm and its application to optimization problems," *Expert Systems with Applications*, vol. 41, no. 15, pp. 6555–6569, 2014.

[28] H. Shah-Hosseini, "Intelligent water drops algorithm: a new optimization method for solving the multiple knapsack

problem," *International Journal of Intelligent Computing & Cybernetics*, vol. 1, no. 2, pp. 193–212, 2008.

[29] Z. Booyavi, E. Teymourian, G. M. Komaki et al., "An improved optimization method based on the intelligent water drops algorithm for the vehicle routing problem," in *Proceedings of the IEEE Symposium on Computational Intelligence in Production and Logistics Systems (CIPLS)*, IEEE, Orlando, FL, USA, December 2015.

[30] S. H. Niu, S. K. Ong, and A. Y. C. Nee, "An improved intelligent water drops algorithm for achieving optimal job-shop scheduling solutions," *International Journal of Production Research*, vol. 50, no. 15, p. 14, 2012.

[31] B. Kallehauge, J. Larsen, O. B. G. Madsen et al., *Vehicle Routing Problem with Time Windows, Column Generation*, Springer, Berlin, Germany, 2005.

[32] I. Kamkar, M. R. Akbarzadeh-T, and M. Yaghoobi, "Intelligent water drops a new optimization algorithm for solving the vehicle routing problem," in *Proceedings of the IEEE International Conference on Systems Man & Cybernetics*, IEEE, Istanbul, Turkey, October 2010.

[33] E. Teymourian, G. M. V. Kayvanfar, and M. Zandieh, "Enhanced intelligent water drops and cuckoo search algorithms for solving the capacitated vehicle routing problem," *Information Sciences*, vol. 334-335, pp. 354–378, 2016.

[34] J. H. Li, H. Guo, Q. H. Zhou, and B. X. Yang, "Vehicle routing and scheduling optimization of ship steel distribution center under green shipbuilding mode," *Sustainability*, vol. 11, no. 15, 2019.

[35] A. Y. Bigazzi and C. Monsere, "Traffic congestion mitigation as an emissions reduction strategy," *Dissertations & Theses—Gradworks*, vol. 278, no. 1–3, pp. 154–158, 2011.

[36] M. M. Solomon, J. Cordeau, H. Etudes et al., *The VRP with Time Windows, The Vehicle Routing Problem*, SIAM Monographs on Discrete Mathematics and Applications, New York, NY, USA, 1999.

[37] http://w.cba.neu.edu/~msolomon/problems.htm.