

Optimization of the Low-Carbon Vehicle Routing Problem with Soft Time Windows: A Hybrid Genetic Algorithm HGA-SIH

Zhilin Zeng

International College, Hunan University of Arts and Science, Changde, China

Abstract: To address the optimization challenge arising from carbon emission constraints and elastic time-sensitive demand in urban last-mile delivery, this paper studies the low-carbon vehicle routing problem with soft time windows (LCVRPTW). A mixed-integer linear programming model is formulated to minimize total system cost, integrating vehicle fixed activation costs, load-dependent fuel consumption and carbon emission costs, and soft time window penalty costs. The problem is proven to be strongly NP-hard. To overcome the limitations of standard genetic algorithms in medium- and large-scale scenarios, a hybrid genetic algorithm (HGA-SIH) is proposed, integrating the sequential insertion heuristic (SIH) with the simulated annealing Metropolis criterion. This paper derives the SIH marginal insertion cost decomposition formula, provides a set-theoretic description of the OX crossover operator, elucidates the statistical mechanics origin of the Metropolis criterion, and establishes temperature schedule convergence conditions. Based on the Markov chain framework, HGA-SIH is proven to converge to the global optimum with probability 1. Time complexity analysis shows per-generation computational complexity of $O(N \cdot n \cdot K)$.

Keywords: Low-Carbon Vehicle Routing Problem; Soft Time Windows; Hybrid Genetic Algorithm; Sequential Insertion Heuristic; Simulated Annealing; Convergence Analysis

1. Introduction

The green and low-carbon transition of the transportation sector, coupled with increasingly refined customer demands for delivery timeliness, has made the superimposition of carbon emission constraints and time window constraints an unavoidable practical issue in urban vehicle routing optimization. [1,2]

Existing research on low-carbon vehicle routing suffers from three major limitations. First, most models treat delivery timeliness using hard time windows, failing to reflect the elastic tolerance characteristic of customer demand. [3-5] Second, emission accounting often uses travel distance as the sole proxy variable, neglecting the significant impact of vehicle load on instantaneous fuel consumption. [6-7] Third, metaheuristic methods such as standard genetic algorithms exhibit low-quality initial solutions and susceptibility to local optima in medium- and large-scale problems, and lack systematic theoretical analysis. [8-10]

To address these gaps, this paper studies the low-carbon vehicle routing problem with soft time windows. [11-15] A comprehensive optimization model is constructed that integrates transportation costs, carbon emission costs, and time window penalty costs, and a load-dependent linear fuel consumption rate function is adopted to improve the accuracy of emission accounting. In terms of solution methods, a hybrid genetic algorithm HGA-SIH is proposed, which combines the sequential insertion heuristic with a simulated annealing mechanism. The algorithm improves the initial population generation strategy, introduces the Metropolis acceptance criterion to modify the mutation stage, and employs an elitism mechanism to enhance solution quality and convergence stability. Theoretical analysis is conducted from both the time complexity and convergence perspectives.

2. Problem Description and Basic Assumptions

2.1 Problem Description

This paper studies the low-carbon vehicle routing problem with soft time windows (LCVRPTW), which is a low-carbon extension variant of the classical VRPTW and has been proven to be NP-hard. [16] The problem is set in an urban last-mile delivery network consisting of

one distribution center and n customer nodes. Each customer has a deterministic cargo demand, service duration, and soft time window. The distribution center is equipped with K vehicles of identical specifications; each vehicle departs from the distribution center, completes its assigned tasks, and returns to the starting point. The decisions include the assignment of customers to vehicles, the visiting sequence and route of each vehicle, and the arrival time at each node. The optimization objective is to minimize the total system cost, which comprises three components: vehicle fixed activation costs, load-dependent transportation and carbon emission costs, and soft time window deviation penalty costs.

2.2 Basic Assumptions

To delineate the model boundaries and account for the characteristics of urban last-mile delivery, the following basic assumptions are made. Customer demand is a static and deterministic parameter with no dynamic fluctuations during the delivery cycle. All vehicles have identical model and performance, with fixed rated load capacity, maximum travel range, and average speed; cargo is loaded only at the distribution center. Each vehicle forms a closed route starting from and returning to the distribution

center; each customer is visited exactly once by exactly one vehicle, and orders cannot be split. The instantaneous fuel consumption rate of a vehicle is a linearly increasing function of its current load, [17-19] and carbon emissions are converted into economic costs through a carbon tax mechanism. Soft time windows are adopted: service within the time window incurs no penalty, early arrival generates waiting costs, [20-21] and late arrival generates delay penalties while service is still permitted; penalties are linearly related to the deviation duration. [22] Road conditions in the delivery area are good, vehicles travel at a constant average speed, [23-24] and dynamic traffic factors such as congestion are not considered.

3. Mathematical Model

3.1 Sets and Parameters

The problem is defined on a complete directed graph $G=(V,A)$, where the node set $V=\{0,1,2,\dots,n\}$ consists of the distribution center (indexed 0) and n customer nodes, and the arc set $A=\{(i,j) \mid i,j \in V, i \neq j\}$ represents the direct travel paths between nodes. The sets and parameters involved in the model are defined in Table 1.

Table 1. Sets and Parameters

Symbol	Description	Unit
V	Set of all nodes, $V=\{0,1,\dots,n\}$, where 0 denotes the distribution center	—
V^+	Set of customer nodes, $V^+ = V \setminus \{0\}$	—
K	Set of available vehicles	—
d_{ij}	Travel distance from node i to node j	km
t_{ij}	Travel time, $t_{ij}=d_{ij}/v$	h
q_i	Cargo demand of customer i	weight unit
s_i	Service duration at customer i	h
$[ET_i, LT_i]$	Expected service time window of customer i	—
Q	Rated load capacity of vehicles	weight unit
$D_{\max}$	Maximum travel range of vehicles	km
v	Average travel speed of vehicles	km/h
c_0	Fixed vehicle activation cost	yuan/vehicle
ρ	Comprehensive fuel unit price including carbon tax (embedding carbon emission factor and carbon tax rate)	yuan/L
α	Fuel consumption rate per unit distance at empty load	L/km
β	Load-dependent fuel consumption rate coefficient	L/(km·weight unit)
c^E, c^L	Unit-time early waiting / late delay penalty cost	yuan/h
M	A sufficiently large positive number (e.g., $M = 10^6$)	—

Table 2. Decision Variables

Variable	Type	Description
x_{ijk}	Binary	Equals 1 if vehicle k travels from node i to node j , 0

3.2 Decision Variables

The decision variables of the model are defined in Table 2.

		otherwise
Y_{ik}	Binary	Equals 1 if customer i is served by vehicle k , 0 otherwise
B_{ik}	Continuous	Time at which vehicle k begins serving customer i
L_{ijk}	Continuous	Current load of vehicle k when traveling from i to j
W_{ik}	Continuous	Waiting time of vehicle k at customer i
U_{ik}	Continuous	Delay time of vehicle k at customer i

3.3 Objective Function

The objective function of the mathematical model minimizes the total system cost, comprising vehicle fixed activation costs, transportation and carbon emission costs, and soft time window penalty costs:

$$\min Z = C_{fixed} + C_{transport} + C_{penalty} \quad (1)$$

(1) Vehicle fixed activation cost: when a vehicle is activated (i.e., serves at least one customer), a fixed cost is incurred:

$$C_{fixed} = c_0 \sum_{k \in K} \sum_{j \in V^+} x_{0jk} \quad (2)$$

(2) Transportation and carbon emission cost: fuel cost and carbon tax cost are merged into a comprehensive transportation cost. The fuel consumption rate of vehicle k on arc (ij) is $f_{ijk} = \alpha + \beta \cdot L_{ijk}$, and the resulting comprehensive transportation cost is: $\alpha + \beta \cdot L_{ijk}$

$$C_{transport} = \rho \sum_{k \in K} \sum_{i \in V} \sum_{j \in V, j \neq i} d_{ij} (\alpha x_{ijk} + \beta \theta_{ijk}) \quad (3)$$

where θ_{ijk} is a linearization auxiliary variable (see Section 3.5), and ρ is the comprehensive fuel unit price including carbon tax, which uniformly incorporates the fuel price, carbon emission factor, and carbon tax rate. $\theta_{ijk} = L_{ijk} \cdot x_{ijk} \rho$

(3) Soft time window penalty cost: when a vehicle arrives earlier than ET_i , a waiting cost is incurred; when it arrives later than LT_i , a delay cost is incurred:

$$C_{penalty} = c^E \sum_{k \in K} \sum_{i \in V^+} W_{ik} + c^L \sum_{k \in K} \sum_{i \in V^+} U_{ik} \quad (4)$$

In summary, the complete objective function is:

$$\begin{aligned} \min Z &= c_0 \sum_{k \in K} \sum_{j \in V^+} x_{0jk} \\ &+ \rho \sum_{k \in K} \sum_{i \in V} \sum_{j \in V, j \neq i} d_{ij} (\alpha x_{ijk} + \beta \theta_{ijk}) \\ &+ c^E \sum_{k \in K} \sum_{i \in V^+} W_{ik} + c^L \sum_{k \in K} \sum_{i \in V^+} U_{ik} \end{aligned} \quad (5)$$

3.4 Constraints

Each customer node is served exactly once by exactly one vehicle:

$$\sum_{k \in K} y_{ik} = 1, \forall i \in V^+ \quad (6)$$

Vehicle flow conservation—a vehicle must depart after arriving at a customer:

$$\sum_{j \in V, j \neq i} x_{ijk} = y_{ik}, \forall i \in V^+, \forall k \in K \quad (7)$$

$$\sum_{j \in V, j \neq i} x_{jik} = y_{ik}, \forall i \in V^+, \forall k \in K \quad (8)$$

Distribution center departure and return constraints (each vehicle is used at most once):

$$\sum_{j \in V^+} x_{0jk} = \sum_{i \in V^+} x_{i0k} \leq 1, \forall k \in K \quad (9)$$

The total load of a vehicle when departing from the distribution center does not exceed the rated capacity:

$$\sum_{i \in V^+} q_i y_{ik} \leq Q, \forall k \in K \quad (10)$$

Upper and lower bound constraints on arc load (linearized using the big-M method):

$$L_{ijk} \leq Q \cdot x_{ijk}, \forall i, j \in V, i \neq j, \forall k \in K \quad (11)$$

$$L_{ijk} \geq \sum_{p \in V^+} q_p y_{pk} - Q(1 - x_{ijk}), \forall i, j \in V, i \neq j, \forall k \in K \quad (12)$$

$$L_{ijk} \geq 0 \quad (13)$$

The total travel distance of a single vehicle in one delivery tour does not exceed the maximum limit:

$$\sum_{i \in V} \sum_{j \in V, j \neq i} d_{ij} x_{ijk} \leq D_{max}, \forall k \in K \quad (14)$$

Linearized representation of waiting time and delay time. Since the penalty terms in the objective function are in the minimization direction and W_{ik} , U_{ik} are non-negative variables, the max function can be exactly represented by the following system of linear inequalities:

$$W_{ik} \geq ET_i - B_{ik}, W_{ik} \geq 0, \forall i \in V^+, \forall k \in K \quad (15)$$

$$U_{ik} \geq B_{ik} - LT_i, U_{ik} \geq 0, \forall i \in V^+, \forall k \in K \quad (16)$$

Service start time is no earlier than the lower bound of the time window:

$$B_{ik} \geq ET_i, \forall i \in V^+, \forall k \in K \quad (17)$$

Time recurrence constraint—lower bound on service start time after vehicle travels from customer i to j :

$$B_{jk} \geq B_{ik} + s_i + t_{ij} - M(1 - x_{ijk}), \forall i \in V^+, \forall j \in V, j \neq i, \forall k \in K \quad (18)$$

Time constraint for vehicle returning to the distribution center:

$$B_{ik} + s_i + t_{i0} \leq D_{max}/v + M(1 - x_{i0k}), \forall i \in V^+, \forall k \in K \quad (19)$$

Variable domain constraints:

$$x_{ijk} \in \{0,1\}, y_{ik} \in \{0,1\} \quad (20)$$

3.5 Linearization Treatment

The objective function (3) contains the nonlinear product term $L_{ijk} \cdot x_{ijk}$. An auxiliary variable $\theta_{ijk} = L_{ijk} \cdot x_{ijk}$ is introduced, and the following constraints are added for exact linearization:

$$\theta_{ijk} \leq Q \cdot x_{ijk}, \theta_{ijk} \leq L_{ijk}, \theta_{ijk} \geq L_{ijk} - Q(1 - x_{ijk}), \theta_{ijk} \geq 0 \quad (21)$$

Derivation: Proof of correctness of the linearization constraints (21)

When $x_{ijk}=0$, from $\theta_{ijk} \leq Q \cdot x_{ijk}=0$ and $\theta_{ijk} \geq 0$, we obtain $\theta_{ijk}=0$, which exactly equals $L_{ijk} \cdot 0$.

$$x_{ijk}=0 \Rightarrow \theta_{ijk} \leq Q \cdot x_{ijk}=0 \Rightarrow \theta_{ijk} \geq 0 \Rightarrow \theta_{ijk}=0 = L_{ijk} \cdot 0$$

When $x_{ijk}=1$, the constraints reduce to $x_{ijk}=1 \Rightarrow \theta_{ijk} \leq Q \cdot x_{ijk}=Q \Rightarrow \theta_{ijk} \geq 0 \Rightarrow \theta_{ijk}=Q = L_{ijk} \cdot 1$. Combining $\theta_{ijk} \leq L_{ijk}$ and $\theta_{ijk} \geq L_{ijk}$ yields $\theta_{ijk}=L_{ijk}$, which exactly equals $L_{ijk} \cdot 1$.

Therefore, for all values of $x_{ijk} \in \{0,1\}$, the constraint system (21) is exactly equivalent to $\theta_{ijk}=L_{ijk} \cdot x_{ijk}$; the linearization is exact rather than approximate.

After this transformation, the transportation cost term in the objective function is rewritten in the form of Equation (3), and the entire model becomes a standard mixed-integer linear program (MILP). Small-scale instances can be solved exactly by commercial solvers such as Gurobi and CPLEX.

3.6 Problem Complexity Analysis

Theorem: LCVRPWTW is strongly NP-hard.

Proof: Consider a special case of LCVRPWTW. Set the carbon emission parameters $\alpha=\beta=\rho=0$ (i.e., disregard all fuel consumption and carbon emission costs), set the soft time window penalty coefficients $c^E=c^L=0$, and enforce $U_{ik}=W_{ik}=0$ for all customers (i.e., prohibit early arrival or delay, reducing to hard time window constraints). Furthermore, let the vehicle rated load capacity Q be a finite constant and the maximum travel range $D_{\max} \rightarrow \infty$ (eliminating the effect of the range constraint).

According to the studies of Solomon (1987) and Desrochers et al. (1988), the decision form of VRPTW is NP-complete and its optimization form is NP-hard. Since the general case of LCVRPWTW contains the above NP-hard special case as a subproblem, the optimization problem of LCVRPWTW is at least as difficult as VRPTW, and is therefore strongly NP-hard.

4. Hybrid Genetic Algorithm Design (HGA-

SIH)

Figure 1 depicts the overall framework of the proposed HGA-SIH. The algorithm starts with a hybrid initial population generated by both the sequential insertion heuristic (Section 4.2) and random permutations, and then iterates through roulette selection, OX crossover with greedy repair, Metropolis-accepted inversion mutation, and elitism replacement until the maximum generation $G_{\max}$ is reached. Details of each component are described in Sections 4.1-4.6.

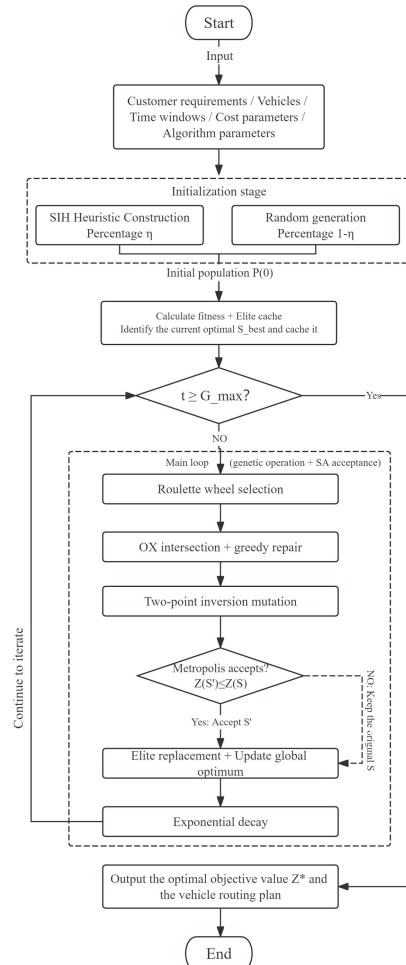


Figure 1. HGA-SIH Overall Flowchart

4.1 Encoding and Decoding Scheme

A natural number split encoding scheme is adopted. A chromosome consists of two parts: a customer visiting sequence and vehicle split markers. Let the total number of customers be n and the number of available vehicles be K ; the total chromosome length is $n+K-1$. The first n positions are a permutation of distinct natural numbers from 1 to n , representing the global visiting order of all customers. The last $K-1$ positions are separator symbols "0", which divide the customer sequence into K consecutive

subsequences. Each subsequence corresponds to the delivery route of one vehicle, and customers within a subsequence are visited in the encoded order.

Formal Description of the Encoding Scheme

Let a chromosome be a vector $c=(c_1, c_2, \dots, c_{n+K-1})$, where the first n components satisfy $\{c_1, \dots, c_n\}=\{1, 2, \dots, n\}$ and are pairwise distinct, and the last $K-1$ components are all 0. Let the positions of the separators in the chromosome be $p_1 < p_2 < \dots < p_{K-1}$, with the conventions $p_0=0, p_K=n+K$. Then the customer sequence corresponding to the route of vehicle k is:

$$R_k=(c_{p_{k-1}+1}, c_{p_{k-1}+2}, \dots, c_{p_k}), k=1, 2, \dots, K$$

The complete route is $0 \rightarrow R_k(1) \rightarrow R_k(2) \rightarrow \dots \rightarrow R_k(|R_k|) \rightarrow 0$, where $|R_k|$ is the number of customers served by vehicle k . If $|R_k|=0$, the vehicle is not activated.

$$0 \rightarrow R_k(1) \rightarrow R_k(2) \rightarrow \dots \rightarrow R_k(|R_k|) \rightarrow 0 |R_k|$$

Encoding example ($n=5, K=2$): The delivery plan corresponding to chromosome $[2, 4, 0, 1, 5, 3]$ is as follows: Vehicle 1 follows the route $0 \rightarrow 2 \rightarrow 4 \rightarrow 0$, and Vehicle 2 follows the route $0 \rightarrow 1 \rightarrow 5 \rightarrow 3 \rightarrow 0$. Figure 2, An example of the split encoding and decoding scheme, $n=5$ customers, $K=2$ vehicles).

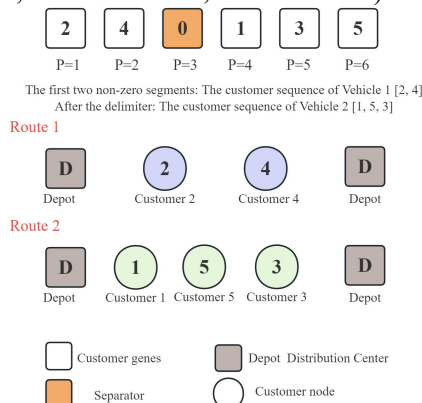


Figure 2. Split Encoding & Decoding Example

This encoding scheme naturally satisfies the constraint that "each customer is visited exactly once," requiring no additional validity check, and can flexibly accommodate route splitting for different numbers of vehicles.

Decoding process: given a chromosome, customer indices are read sequentially and assigned to the current vehicle route. Upon encountering the separator "0", the decoder switches to the next vehicle until all customers are assigned. Each vehicle departs from the distribution center, visits customers in the assigned order, and returns to the distribution center.

4.2 Initial Population Improvement: Sequential Insertion Heuristic (SIH)

Standard genetic algorithms generate the initial population in a completely random manner, [8] and a large number of low-quality individuals slow down convergence. This paper adopts a hybrid generation strategy of "SIH heuristic solutions + random solutions," where a proportion η of individuals are generated by the sequential insertion heuristic and a proportion $1-\eta$ are randomly generated, balancing initial solution quality and population diversity.

The sequential insertion heuristic inserts customers one by one into the optimal position of the current route based on the criterion of minimizing cost increment. Let the current route be $P=(0, v_1, \dots, v_m, 0)$. When inserting customer i between nodes u and v , the original arc (u, v) is replaced by (u, i) and (i, v) . The transportation and carbon emission cost increment is:

$$\Delta C_{\text{trans}} = \rho [d_{ui}(\alpha + \beta L_{uv}) + d_{iv}(\alpha + \beta(L_{uv} - q_i)) - d_{uv}(\alpha + \beta L_{uv})] \quad (22)$$

where L_{uv} is the load on arc (u, v) . The first two terms represent the cost increment due to the change in travel distance, and the third term represents the cost saving due to the load reduction after customer delivery. The time window penalty increment consists of the penalty for customer i itself and the time-shift penalty for customers after the insertion position; a local rescheduling strategy is adopted to recompute only the penalty changes after the insertion position. The marginal total insertion cost is:

$$\Delta C_{\text{trans}} = \rho [\alpha(d_{ui} + d_{iv} - d_{uv}) + \beta L_{uv}(d_{ui} + d_{iv} - d_{uv}) - \beta q_i d_{iv}] \quad (23)$$

The execution steps of SIH are as follows. Initialize all vehicle routes to contain only the distribution center. Sort customers in ascending order of the earliest time window start time ET_i . Sequentially extract each customer, traverse all adjacent node pairs in all routes, compute the insertion cost according to Equation (23), and select the feasible position with the minimum cost for insertion (feasibility means that the load after insertion does not exceed Q and the range

does not exceed D_{max}). If none of the existing routes can accommodate the customer, activate a new vehicle. Repeat until all customers are inserted. SIH checks constraints at each insertion step; by mathematical induction, the final generated solution satisfies all load and range constraints, and flow conservation is naturally guaranteed by the encoding scheme.

Initial individuals generated through SIH possess superior route structures and cost levels, effectively raising the overall starting quality of the population and shortening the convergence period. Retaining a proportion $1 - \eta$ of random individuals ensures population diversity and prevents the algorithm from prematurely converging to local optima.

4.3 Fitness Function and Selection Operator

Fitness function: the total cost of the objective function defined in Chapter 3 (Equation (5)) is directly used as the evaluation metric. Since the optimization objective is minimization, the fitness value is defined as:

$$\text{fitness}(S) = \frac{1}{Z(S) + \epsilon} \quad (24)$$

where $\epsilon = 10^{-6}$ is a small positive number to prevent division by zero, and S represents an individual (i.e., a complete delivery plan). A larger fitness value indicates a better individual.

Properties of the Fitness Function

Since $Z(S) \geq 0$ holds for all feasible solutions and $f(x) = 1/(x + \epsilon)$ is a strictly monotonically decreasing function on $x \geq 0$, we have $\text{fitness}(S_1) > \text{fitness}(S_2)$ if and only if $Z(S_1) < Z(S_2)$. This guarantees that the fitness ranking is completely consistent with the objective function ranking, introducing no ranking bias.

Furthermore, $\text{fitness}(S) \in (0, 1/\epsilon]$ is a bounded function, which guarantees that the sum of probabilities in roulette wheel selection equals 1 and all probabilities are non-negative.

Selection strategy: a combined strategy of roulette wheel selection and elitism is adopted. The probability of an individual being selected is:

$$P_i = \frac{\text{fitness}_i}{\sum_{j=1}^N \text{fitness}_j} \quad (25)$$

where N is the population size. Roulette wheel selection balances selection pressure and population diversity; the elitism strategy (see Section 4.6) directly copies the best individual of the previous generation to the next generation, preventing superior route structures from being lost during evolution.

4.4 Crossover Operator: Order Crossover (OX) and Repair Strategy

The order crossover operator (OX) is employed for crossover operations. [19] Its core advantage is that it effectively preserves the relative order information of customer visits in the parent chromosomes. The specific operation is as follows: two crossover points are randomly

selected to determine a crossover interval; the offspring retains the customer sequence of parent 1 within this interval, and the remaining positions are filled with customers not yet appearing in the relative order of parent 2. The other offspring is generated symmetrically. Since customers within the interval are pairwise distinct and customers filled in the remaining positions do not belong to the interval set, the offspring generated by OX crossover is necessarily a permutation of the customer set.

Offspring generated by crossover may violate vehicle load constraints. This paper adopts a greedy repair mechanism: for each overloaded route in the offspring, the customer with the largest marginal penalty cost is removed and inserted into the feasible position with the minimum cost increment in another non-overloaded route (insertion cost computed according to Equation (23)). If insertion into any existing route is impossible, a new vehicle is activated. If the total number of vehicles after repair exceeds K , the offspring is rejected and crossover is re-executed (up to 3 retries; if still unsuccessful, the parent is directly copied).

4.5 Mutation Operator Improvement

4.5.1 Statistical mechanics origin of the metropolis criterion

The Metropolis criterion originates from the canonical ensemble theory in statistical mechanics. [20] At temperature T , the probability that a system is in a microstate with energy E follows the Boltzmann distribution:

$$P(E) = \frac{1}{Z(T)} \exp\left(-\frac{E}{k_B T}\right) \quad (26)$$

where k_B is the Boltzmann constant and $Z(T) = \sum_E \exp(-E/k_B T)$ is the partition function.

When the system transitions from state S (energy E) to state S' (energy E'), if $E' \leq E$, the transition necessarily occurs (the system tends toward a lower energy state); if $E' > E$, the transition probability is the ratio of the Boltzmann probabilities of the two states:

$$P(S \rightarrow S') = \frac{P(E')}{P(E)} = \frac{\exp\left(-\frac{E'}{k_B T}\right)}{\exp\left(-\frac{E}{k_B T}\right)} = \exp\left(-\frac{E' - E}{k_B T}\right) \quad (27)$$

In optimization problems, by analogizing the objective function value to system energy and the control parameter to temperature (absorbing k_B), the Metropolis acceptance criterion is

obtained:

$$P_{accept} = \begin{cases} 1 & Z(S') \leq Z(S) \\ \exp(-\frac{Z(S') - Z(S)}{T_t}) & Z(S') > Z(S) \end{cases} \quad (28)$$

4.5.2 Mutation operation and temperature schedule

Mutation operation: two-point inversion mutation is adopted. Two positions r_1 and r_2 ($1 \leq r_1 < r_2 \leq n$) are randomly selected in the chromosome, and the customer visiting sequence between these two positions is completely reversed, while the remaining parts and separator positions remain unchanged. The inversion operation does not change the total number of vehicles or the customer allocation, but only adjusts the visiting order.

Mathematical Description of Two-Point Inversion Mutation

Let the chromosome customer sequence be $c = (c_1, \dots, c_n)$ and the mutated sequence be c' . Then:

$$c'_i = \begin{cases} c_{r_1+r_2-i}, & r_1 \leq i \leq r_2 \\ c_i, & \text{otherwise} \end{cases}$$

Since inversion is a permutation transformation on $\{1, \dots, n\}$ (an involution, i.e., applying it twice restores the original), the mutated sequence remains a valid permutation.

Temperature Schedule

The temperature adopts an exponential decay strategy:

$$T_{t+1} = T_t \cdot \alpha, T_0 > 0, 0 < \alpha < 1 \quad (29)$$

Its closed-form solution is $T_t = T_0 \cdot \alpha^t$

Convergence Conditions of the Temperature Schedule

From Equation (29),

$$\lim_{t \rightarrow \infty} T_t = T_0 \cdot \lim_{t \rightarrow \infty} \alpha^t = 0 \quad (\text{since } 0 < \alpha < 1).$$

Therefore, as the evolution generation $t \rightarrow \infty$:

$$\lim_{t \rightarrow \infty} P_{accept}(Z' - Z) = \lim_{t \rightarrow \infty} \exp(-\frac{Z' - Z}{T_0 \alpha^t}) = 0 \quad \text{Th}$$

at is, in the limiting case, the algorithm degenerates to pure greedy search (accepting only better solutions), guaranteeing eventual convergence. In the early stages of evolution, T_t is relatively large and the probability of accepting inferior solutions is high, which is beneficial for global exploration. This "exploration first, exploitation later" scheduling strategy is the core idea of simulated annealing algorithms.

It should be noted that classical simulated annealing theory requires the temperature to satisfy a logarithmic decay condition $T_t \geq c/\ln(1+t)$ to guarantee convergence to the

global optimum with probability 1. [21] Although exponential decay does not satisfy this theoretical condition, it exhibits good convergence performance in practice and offers higher computational efficiency.

In the early stages of evolution, T_t is high and $\exp(-\Delta/T_t)$ is large, so the algorithm accepts inferior solutions with a relatively high probability, thereby extensively exploring the solution space globally and avoiding being trapped in unfavorable regions due to initial population bias. As the number of evolution generations increases, T_t gradually decreases, the probability of accepting inferior solutions approaches zero, and the algorithm gradually transitions to greedy search, ensuring eventual convergence to high-quality solutions. This mechanism achieves a smooth transition between global exploration and local exploitation.

The mutation operation is executed after selection and crossover. Each individual determines whether to undergo mutation with mutation probability p_m . The Metropolis acceptance criterion judges whether the mutated individual enters the next-generation population, rather than determining the trigger probability of the mutation operation itself—this ensures that every individual that undergoes mutation in each generation passes through quality evaluation and screening.

4.6 Elitism Strategy

To prevent crossover and mutation operations from destroying superior individuals, this paper adopts a strict elitism strategy:

Before each generation of evolution, identify the best individual S_{best} from the current population (i.e., the individual with the minimum Z) and cache it.

After completing selection, crossover, and mutation operations, replace the worst individual in the new population with the cached S_{best} (if S_{best} already exists in the new population, replace the second-worst individual).

Meanwhile, maintain a global best solution S_{global} , updating it whenever $Z(S_{best}) < Z(S_{global})$.

Monotonic Convergence of the Elitism Strategy
Let $Z_t = \min_{S \in P_t} Z(S)$ be the best objective value of the population at generation t . Since the elitism strategy directly copies the best individual S_{best}^t of generation t into the

population of generation $t+1$, we have $Z_{t+1} \leq Z(S_{best}^t) = Z_t$. That is, $Z_{t=0}^\infty$ is a non-increasing sequence.

Furthermore, since $Z_t \geq Z_{\min} > 0$ (the objective function is bounded below), by the monotone convergence theorem, $\lim_{t \rightarrow \infty} Z_t$ exists. This guarantees that the best fitness of the algorithm does not deteriorate during evolution, i.e., it possesses monotonic convergence.

This strategy guarantees that the algorithm has monotonic convergence—the best fitness of the population is non-decreasing during evolution (corresponding to non-increasing objective cost), effectively preventing the destructive effects of genetic operations.

5. Algorithm Theoretical Analysis

5.1 Time Complexity Analysis

Let the total number of customers be n , the number of vehicles be K , the population size be N , and the maximum number of iterations be $G_{\max}$. The time complexities of the core operators are as follows. Fitness computation for a single individual is $O(n)$, totaling $O(Nn)$ per generation. Roulette wheel selection is $O(N^2)$ (can be optimized to $O(N \log N)$ using cumulative probability plus binary search). A single OX crossover is $O(n)$, totaling $O(Nn)$ per generation. A single two-point inversion mutation is $O(n)$, totaling $O(Nn)$ per generation. Elitism is $O(n)$. In summary, the per-generation time complexity is $O(N^2 + Nn)$, which simplifies to $O(Nn)$ when $N \leq n$. SIH initial population generation is executed only once during the initialization phase, with complexity $O(\eta N(n^2 + nK))$. The overall time complexity of the algorithm is:

$$T_{\text{total}} = O(\eta N(n^2 + nK)) + O(G_{\max} \cdot N \cdot n \cdot K) \quad (30)$$

In practical applications, N and K are typically much smaller than n , and $G_{\max}$ is a preset constant; the computational efficiency of the algorithm can meet the requirements of medium- and large-scale scenarios.

5.2 Convergence Analysis Based on Markov Chains

Let the solution space Ω be the set of all chromosomes satisfying the encoding rules, and the population space $\Lambda = \Omega^N$. The population sequence P_t of HGA-SIH constitutes a finite-state Markov chain, [22] whose state transitions are jointly determined by the four operators: selection, crossover, mutation, and elitism.

When the mutation probability $p_m > 0$ and the crossover probability $p_c > 0$, two-point inversion mutation can transform any individual into any other individual through a finite number of transpositions, and the Metropolis criterion accepts each mutation with positive probability; therefore, the Markov chain is irreducible. Meanwhile, since $p_m < 1$ and $p_c < 1$, the probability of remaining in the original state after one step transition from any state is positive, so the Markov chain is aperiodic. The elitism strategy guarantees that the best individual of each generation is directly copied to the next generation; therefore, the optimal population set $\Lambda = P \in \Lambda: \min_{S \in P} Z(S) = Z^*$ is a closed set of the Markov chain, which, once entered, cannot be left.

For an irreducible and aperiodic finite Markov chain, there exists a unique stationary distribution and convergence to this distribution from any initial state. Since Λ is a closed set, the stationary distribution has zero probability mass outside Λ , and therefore:

$$\lim_{t \rightarrow \infty} \Pr(\min_{S \in P_t} Z(S) = Z^*) = 1 \quad (31)$$

That is, the HGA-SIH algorithm converges to the global optimum with probability 1. The mutation operator guarantees global reachability of the state space, and elitism guarantees that once the optimal solution is found it is not lost; the combination of the two is the key to convergence. The simulated annealing Metropolis criterion regulates the balance between exploration and exploitation through temperature scheduling, enabling the algorithm to extensively explore the solution space in the early stages of evolution.

5.3 Discussion of Algorithm Properties

Synthesizing the above analysis, HGA-SIH possesses three noteworthy properties. First, there is a trade-off between global convergence and convergence speed. The convergence proof is an asymptotic result, and the actual convergence speed within finite iterations depends on parameter settings; a higher initial temperature and slower cooling rate favor global exploration but result in slower convergence. Second, SIH solutions and random solutions are complementary: SIH provides a high-quality search starting point, [23] while random solutions maintain population diversity; the mixture of the two avoids both population homogenization caused by complete SIH and low-quality starting points caused by complete

randomization. Third, the Metropolis criterion and elitism form an "exploration-protection" synergistic mechanism: the former allows inferior solutions to be accepted with a certain probability to enhance the ability to escape local optima, while the latter guarantees that the best individual of each generation is not destroyed by genetic operations.

6. Conclusion

This paper studies the low-carbon vehicle routing problem with soft time windows. A comprehensive optimization model is constructed that integrates transportation costs, carbon emission costs, and time window penalty costs. A load-dependent linear fuel consumption rate function is adopted to improve the accuracy of emission accounting, and the model is transformed into a standard mixed-integer linear program by introducing auxiliary variables. In terms of solution methods, a hybrid genetic algorithm HGA-SIH is proposed, which integrates the sequential insertion heuristic with a simulated annealing mechanism: the initial population generation strategy is improved through SIH to enhance starting point quality, the Metropolis acceptance criterion is introduced to modify the mutation stage and enhance the ability to escape local optima, and an elitism mechanism is adopted to ensure that the optimal solution is not lost. Theoretical analysis shows that the per-generation time complexity of the algorithm is $O(Nn)$, and the overall complexity is linearly related to the number of customers. Based on the Markov chain framework, the algorithm is proven to converge to the global optimum with probability 1. Future research can further extend to scenarios closer to actual operations, such as stochastic demand, multi-depot collaborative scheduling, and dynamic traffic networks.

Acknowledgments

This work was supported by the National College Students' Innovation and Entrepreneurship Training Program under Grant No. 202510549002 (Project Title: Design of Intelligent Express Delivery Cluster Robots in Discrete Manufacturing Scenarios).

References

[1] Yin, N. (2023). Multiobjective Optimization for Vehicle Routing Optimization Problem in Low-Carbon Intelligent Transportation.

IEEE Transactions on Intelligent Transportation Systems, 24(11), 13161-13170.

- [2] Liu, Y., Yu, Y., Baldacci, R., Tang, J., & Sun, W. (2025). Optimizing carbon emissions in green logistics for time-dependent routing. *Transportation Research Part B-Methodological*, 192, 103155.
- [3] Boz, E., Calik, A., & Sahin, Y. (2024). Solution approaches for the green vehicle routing problem with time window and simultaneous pickup and delivery. *Journal of the Faculty of Engineering and Architecture of Gazi University*, 39(2), 757-770.
- [4] Gao, J., Zheng, X., Gao, F., Tong, X., & Han, Q. (2022). Heterogeneous Multitype Fleet Green Vehicle Path Planning of Automated Guided Vehicle with Time Windows in Flexible Manufacturing System. *Machines*, 10(3), 197.
- [5] Wang, D. L., Ding, A., Chen, G. L., & Zhang, L. (2023). A combined genetic algorithm and A search algorithm for the electric vehicle routing problem with time windows. *Advances in Production Engineering & Management*, 18(4), 403-416.
- [6] Lai, D., Costa, Y., Demir, E., Florio, A. M., & Van Woensel, T. (2024). The pollution-routing problem with speed optimization and uneven topography. *Computers & Operations Research*, 164, 106557.
- [7] Li, W., Sun, Q., Guo, S., & Ma, H. (2025). Vehicle route and speed optimization with carbon emission control in fragile goods transportation. *International Journal of General Systems*.
- [8] Zhu, J. (2022). Solving Capacitated Vehicle Routing Problem by an Improved Genetic Algorithm with Fuzzy C-Means Clustering. *Scientific Programming*, 2022, 8514660.
- [9] Zhang, W., Li, H., Yang, W., Zhang, G., & Gen, M. (2022). Hybrid multiobjective evolutionary algorithm considering combination timing for multi-type vehicle routing problem with time windows. *Computers & Industrial Engineering*, 171.
- [10] Zhou, L., Zhang, H., & Fu, J. (2025). Green vehicle routing optimization based on dynamic constraint selection co-evolutionary algorithm. *Scientific Reports*, 15(1), 17688.
- [11] Jie, K.-W., Liu, S.-Y., & Sun, X.-J. (2022).

- A hybrid algorithm for time-dependent vehicle routing problem with soft time windows and stochastic factors. *Engineering Applications of Artificial Intelligence*, 109, 104606.
- [12] He Meiling, Yang Mei, Wu Xiaohui. Research on the Vehicle Routing Problem with Simultaneous Pickup and Delivery and Soft Time Windows from a Low-Carbon Perspective. *Journal of Jiangsu University (Natural Science Edition)*, 2026, 47(3): 283–291.
- [13] Ge Xianlong, Zhu Ziqiang. Electric Vehicle Routing Optimization Problem with Soft Time Windows. *Industrial Engineering and Management*, 2019, 24(4): 96–104+112.
- [14] Xie Yong, Hu Rong, Qian Bin, Chen Shaofeng, Zhang Guilian, Zhang Xiaodi. An Improved Population-Based Incremental Learning Algorithm for Solving the Vehicle Routing Optimization Problem with Soft Time Windows. *Journal of Nanjing University of Science and Technology*, 2016, 40(1): 110–116.
- [15] Yang Jin, Ma Liang. Application of Bee Colony Optimization Algorithm in Vehicle Routing Problem with Soft Time Windows. *Forecasting*, 2010, 29(6): 67–70+61.
- [16] Ren, T., Luo, T., Jia, B., Yang, B., Wang, L., & Xing, L. (2023). Improved ant colony optimization for the vehicle routing problem with split pickup and split delivery. *Swarm and Evolutionary Computation*, 77, 101228.
- [17] Fang, W., Guan, Z., Yue, L., Zhang, Z., Wang, H., & Meng, L. (2022). Heterogeneous-vehicle distribution logistics planning for assembly line station materials with multiple time windows and multiple visits. *International Journal of Industrial Engineering Computations*, 13(4), 473-490.
- [18] Hien, V. Q., Dao, T. C., & Binh, H. T. T. (2023). A greedy search based evolutionary algorithm for electric vehicle routing problem. *Applied Intelligence*, 53(3), 2908-2922.
- [19] Liu, L., Gu, R., Huo, J., & Zhou, Y. (2023). Origin-Oriented Shuffled Frog Leaping Vehicle Routing Multiobjective Optimization Algorithm. *Journal of Database Management*, 34(3).
- [20] Dong, Y., Wang, S., & Liu, X. (2025). Flow Shop Scheduling with Limited Buffers by an Improved Discrete Pathfinder Algorithm with Multi-Neighborhood Local Search. *Processes*, 13(8), 2325.
- [21] Niu, Y., Yu, F., Yao, H., & Yang, Y. (2022). Multi-equipment coordinated scheduling strategy of U-shaped automated container terminal considering energy consumption. *Computers & Industrial Engineering*, 174, 108804.
- [22] Zeng, Z., Ding, J. (2025). Design of an Intelligent Courier Vehicle System Based on STM32 and Free RTOS. *Electronic Making*, 2025, 33(15): 7-12.
- [23] Zhu, Y., Li, C., & Lee, K. Y. (2022). The NR-EGA for the EVRP Problem with the Electric Energy Consumption Model. *Energies*, 15(10), 3681.
- [24] Yue, B., Ma, J., Shi, J., & Yang, J. (2024). A Deep Reinforcement Learning-Based Adaptive Search for Solving Time-Dependent Green Vehicle Routing Problem. *Ieee Access*, 12, 33400-33419.