Dynamic Constraint-Aware Particle Swarm Optimization for Resource Allocation in Logistics and Transportation

Xijing Ou
School of Economics and Management, Jiaozuo University, Jiaozuo 454000, Henan, China
E-mail: ouxijing827@163.com

Keywords: logistics and transportation, particle swarm optimization algorithm, resource allocation, scheduling optimization, transportation cost

Received: May 13, 2025

With the rapid expansion of the logistics and transportation industry, effective resource allocation and scheduling have become critical to operational efficiency. This study proposes a Constraint-Aware Particle Swarm Optimization model for Logistics Resource Allocation and Scheduling (LRAS-PSO), which incorporates three core innovations: (1) a Transportation Scenario Complexity Index (TSCI) to enable adaptive parameter tuning, (2) a real-time monitoring module utilizing IoT data, and (3) a constraint-handling mechanism to address emergencies like vehicle failure and route blockage. The model is empirically evaluated on the widely used Solomon benchmark dataset. Compared with traditional linear programming, genetic algorithms, and ant colony optimization, LRAS-PSO demonstrates a minimum 25.5% reduction in transportation cost and approximately 15–20% improvement in transportation efficiency across multiple logistics scenarios. These results underscore the practical value of LRAS-PSO in enabling intelligent and adaptive logistics management.

Povzetek: Predstavljen je izboljšan algoritem rojev delcev (LRAS-PSO) za razporejanje virov v logistiki, ki z dinamično prilagoditvijo parametrov in obvladovanjem omejitev izboljša učinkovitost in zmanjša stroške.

1 Introduction

In today's highly information-based and highly commercialized era, the importance of the logistics and transportation industry is beyond elaboration. According to incomplete statistics, the value of goods transported through logistics and transportation around the world each year is as high as hundreds of trillions. For example, in a certain large economic region, the value of goods involved in logistics and transportation each year is tens of trillions, and this value continues to rise at a rate of about 8% - 12% per year [1]. Behind this huge number is the survival and development support of countless companies, and it is also the key support for the stable operation of the global economy [2].

However, there have always been many serious problems in the allocation and scheduling of logistics and transportation resources. For example, in the past year, a well-known logistics company had thousands of delayed deliveries due to unreasonable resource allocation and scheduling errors, involving tens of thousands of tons of goods and direct economic losses estimated to be hundreds of millions of yuan [3]. These delayed deliveries not only caused strong dissatisfaction among many customers, causing the company's customer churn rate to increase by about 15%, but also triggered a chain reaction in the industry, affecting the normal operation rhythm of many companies in the relevant industrial chain [4].

From a macro perspective, the entire logistics and transportation industry generally lacks scientific and effective means for resource allocation and scheduling. Traditional manual allocation and scheduling methods are often too subjective and inefficient, making it difficult to cope with the increasingly complex and changing logistics and transportation needs [5]. For example, when faced with sudden large-scale order growth or temporary changes in transportation routes, manual methods often require a lot of time and energy to re-plan, and the probability of error is extremely high. According to relevant surveys, the average error rate of manual scheduling is around 20%-30% [6]. This low efficiency and high error rate have seriously restricted the development of the logistics and transportation industry and have also had a huge negative impact on the efficiency of economic operations [7].

At present, there are many relevant research results in the field of logistics transportation resource allocation and scheduling. Many scholars and research institutions are trying to optimize this process through various algorithms and models. For example, some researchers use traditional linear programming algorithms to optimize resource allocation, which improves the rationality of resource allocation to a certain extent. However, this algorithm often seems to be unable to cope with complex nonlinear logistics transportation problems, and its optimization effect is very limited. The cost reduction is only about 10% - 15% at most [8].

Some studies have also adopted intelligent algorithms such as genetic algorithms, which have made some progress compared to traditional algorithms and can adapt to complex logistics and transportation environments to a certain extent, and can improve transportation efficiency by about 20%-25%. However, genetic algorithms have defects such as high computational complexity and easy to fall into local optimal solutions. In practical applications, they cannot fully meet the needs of logistics and transportation companies for efficient and accurate resource allocation and scheduling.

Among many studies, the particle swarm optimization (PSO) algorithm has gradually attracted attention. However, most of the current research based on the PSO algorithm is still in the theoretical stage. When it is actually applied to the allocation and scheduling of logistics transportation resources, there are still many problems that have not been effectively solved. For example, the parameter setting in the PSO algorithm lacks a unified standard. Different parameter settings will lead to very different results. The stability of its optimization effect is difficult to guarantee. In addition, the existing research based on the PSO algorithm does not consider some special situations and constraints in logistics transportation in a comprehensive manner, which greatly reduces its applicability in practical applications.

This paper aims to build a more suitable optimization model for logistics and transportation resource allocation and scheduling by deeply studying the particle swarm optimization (PSO) algorithm and making targeted improvements and perfections to it. By reasonably setting the algorithm parameters and taking into full consideration various practical constraints, the stability and applicability of the algorithm can be improved, thus achieving efficient, reasonable allocation and precise scheduling of logistics and transportation resources.

The innovation of this study lies in breaking through the limitations of the previous application of PSO algorithm in this field. .By introducing new parameter adjustment strategies and constraint handling mechanisms, the LRAS-PSO model aims to improve transportation performance. Experimental benchmarks demonstrate notable cost reductions of up to 25.5% and efficiency gains of 15–20%, positioning LRAS-PSO as a competitive tool for enhancing logistics operations under dynamic and constrained environments.

The primary research objective of this study is to develop a Particle Swarm Optimization (PSO)-based model equipped with adaptive parameter tuning and real-time constraint handling capabilities. The model aims to minimize overall transportation costs and maximize delivery efficiency under complex and dynamic logistics conditions. By integrating scenario complexity evaluation, iterative learning strategies, and adaptive response mechanisms, the proposed model targets large-scale practical deployments in heterogeneous logistics networks.

The motivation for this work stems from the growing complexity of logistics systems, where traditional scheduling methods often fail to achieve optimal performance in the presence of real-time constraints, dynamic events, and large-scale data. To address these challenges, this paper proposes the LRAS-PSO model, which introduces three core contributions:

(1) a scenario complexity index to guide adaptive parameter tuning,

(2) a real-time monitoring and constraint handling mechanism for exceptional transport events, and

(3) an optimization model tailored for minimizing total transportation cost while maintaining high delivery efficiency.

The remainder of this paper is organized as follows: Section 2 reviews related work and identifies key limitations. Section 3 details the proposed LRAS-PSO model. Section 4 presents experimental evaluations and results. Section 5 concludes the paper with discussions on limitations and future directions.

2 Literature review

2.1 Application status of related algorithms in logistics and transportation optimization

Traditional linear programming algorithms have been used for a long time in the optimization of logistics and transportation resource allocation. Studies have shown that they can optimize resource allocation to a certain extent and make resource utilization more reasonable, but the algorithm itself has great limitations. In the face of complex logistics and transportation scenarios, especially those with nonlinear characteristics, its optimization ability is severely weakened, and the cost reduction that can be achieved is only between 10% and 15%. This limited optimization effect makes it difficult to meet the urgent needs of the modern logistics and transportation industry for efficient resource allocation and scheduling. Moreover, in the case of large-scale logistics and transportation data and complex transportation networks, its computational efficiency has become very low, passively leading to delays in the overall logistics and transportation decision-making process.

As an intelligent algorithm, genetic algorithm has also been tried in the field of logistics and transportation optimization. Compared with the traditional linear programming algorithm, it has made some progress and can adapt to some complex logistics and transportation environments, and can improve transportation efficiency by about 20% - 25% [8]. However, its own defects are also very obvious. The high computational complexity makes it take too long to process a large amount of logistics and transportation data. The characteristic of being prone to local optimal solutions makes its optimization results often not global optimal. Therefore, in practical applications, it cannot well meet the needs of logistics and transportation companies for efficient and accurate resource allocation and scheduling. In actual logistics and transportation business, due to these defects, its optimization effect is greatly reduced in some complex transportation tasks, which passively affects the willingness of logistics and transportation companies to adopt it [9].

The PSO algorithm has gradually attracted attention in the field of logistics and transportation resource allocation and scheduling, but its application is still mostly in the theoretical stage. There is a lack of unified standards for its parameter settings, and the results under different parameter settings vary greatly, which makes it difficult to ensure the stability of its optimization effect [10]. For example, in a set of simulated logistics and transportation tasks, a slight change in parameter settings can cause the fluctuation of transportation costs to reach more than 20%. This instability seriously restricts its application in actual logistics and transportation. In addition, existing research based on the PSO algorithm does not consider special situations and constraints in logistics and transportation, such as temporary failures of transportation vehicles and special weather affecting transportation routes, resulting in its poor applicability in practical applications. In many actual logistics and transportation scenarios, it cannot effectively play its optimization role, which passively makes it subject to many obstacles in actual promotion.

The comparison of optimization algorithms for logistics scheduling is shown in Table 1.

Table 1: Comparison of optimization algorithms for logistics scheduling

Algorithm	Optimization Quality	Scalability	Adaptability to Constraints	Real-world Applicability
Linear Programming	Low (10–15% cost reduction)	Poor	Poor	Limited
Genetic Algorithm	Moderate (20–25%)	Medium	Moderate	Moderate
Ant Colony Optimization	Moderate	Medium	Moderate	Moderate
Standard PSO	Moderate to High	Medium	Low	Experimental
LRAS-PSO (Proposed)	High (≥30%)	High	High	Strong Practicality

2.2 Analysis of key issues in PSO algorithm application in logistics and transportation

The parameter setting of the PSO algorithm has a huge impact on its optimization effect. However, there is currently no widely recognized standard parameter setting method. In different logistics and transportation scenarios, due to factors such as transportation distance, cargo volume, and number of transportation nodes, the appropriate parameter values are also very different [11]. In an internal test of a logistics and transportation company, it was found that when the transportation distance was in the range of 500-1000 kilometers and the cargo volume was in the range of 1000-2000 tons, a certain set of parameter settings could reduce the transportation cost by 15%. However, when the transportation distance and cargo volume changed significantly, the same parameter setting could cause the transportation cost to increase instead of decrease, up to 10% [12]. The uncertainty of this parameter setting and its inadaptability to different scenarios greatly limit the effective application of the PSO algorithm in the allocation and scheduling of logistics and transportation resources.

There are many special situations and constraints in the process of logistics transportation, such as temporary control of transportation roads, sudden failure of transportation vehicles, etc. Most of the existing research based on PSO algorithm does not fully consider these factors, resulting in that in practical applications, once these special situations are encountered, the resource allocation and scheduling solutions generated by the algorithm often fail [13]. According to statistics, in logistics transportation tasks with special circumstances, the solutions generated based on imperfect PSO algorithms cannot be effectively executed, and about 40% of the tasks need to be manually readjusted. This not only increases labor costs, but may also lead to issues such as delayed delivery of goods, reducing the overall efficiency of logistics transportation.

2.3 Improvement Direction and Future Outlook

In order to make the PSO algorithm better applied to the allocation and scheduling of logistics transportation resources, it is necessary to first establish a scientific and reasonable parameter setting system. Through the analysis and simulation experiments of a large amount of actual logistics transportation data, the parameter setting rules under different transportation scenarios are summarized to improve the accuracy and stability of parameter settings [14]. Secondly, the ability to handle special situations and constraints should be strengthened [15]. Introducing more real-time data monitoring and feedback mechanisms will enable the algorithm to perceive the occurrence of special situations in a timely manner and automatically adjust the resource allocation and scheduling plan, thereby enhancing its applicability in practical applications [16].

With the continuous development of the logistics and transportation industry and the increasing demand for intelligence and efficiency, the application prospects of the PSO algorithm in this field are still broad. If the key problems currently existing can be effectively solved, the PSO algorithm is expected to become one of the core optimization algorithms for logistics and transportation resource allocation and scheduling. It is expected that in the next 3-5 years, the improved PSO algorithm can reduce logistics and transportation costs by at least 30% and improve transportation efficiency by more than 35%, bringing significant economic benefits to logistics and transportation companies, promoting the entire logistics and transportation industry to develop in a smarter and more efficient direction, and passively promoting the logistics and transportation industry to further strengthen its supporting role in the global economy [17].

3. Research methods

3.1 Theoretical basis expansion of particle swarm optimization algorithm

The particle swarm optimization (PSO) algorithm is an optimization algorithm based on swarm intelligence, which is inspired by the simulation of bird flock foraging behavior. In this algorithm, through the collaboration and information sharing between particles, the optimal solution to complex problems can be effectively explored.D dimensional search space, where there exists an For particles$i$ In terms of$t$ At the iteration, its velocity vector $\mathbf{v}_i^t=(v_{i1}^t,v_{i2}^t,\cdots,v_{iD}^t)$ With position vector $\mathbf{x}_i^t=(x_{i1}^t,x_{i2}^t,\cdots,x_{iD}^t)$ The update of follows strict mathematical rules. The specific update formula is shown in Formula 1 [18] .

$$v_{id}^{t+1}=\omega v_{id}^t+c_1r_{1id}^t(p_{id}^t-x_{id}^t)+c_2r_{2id}^t(g_{id}^t-x_{id}^t) \quad \text{for } d=1,2,\cdots,D \ (1)$$

in, ω Represents the inertia weight, which is used to balance the global and local search capabilities of particles;c_1 andc_2 It is called the learning factor, which guides the particles to learn from their own historical optimal position and the global optimal position respectively; r_{1id}^t and r_{2id}^t is between 0 arrive 1 They introduce a certain degree of randomness into the particle search process, preventing particles from falling into the local optimum too early.p_{id}^t Represents particles$i$ In the$t$ The iteration$d$ The best historical position of Wei,g_{id}^t This means that the entire population t The iteration d The global optimal position of the dimension [19] .

To better understand inertia weight ω , learning factorc_1 andc_2 We can analyze the impact of speed update in more detail. In the early iteration of the algorithm, in order to enable particles to quickly search for potential optimal solutions in a larger range, it is necessary to ω The value of skewness is biased towards larger values to enhance the global search capability. In the later stages of the iteration, as the search range gradually narrows, in order to find the optimal solution more accurately, ω should be gradually reduced to focus on local search. This process can be explained by formula 2.

$$\omega=\omega_{\text{start}}-(\omega_{\text{start}}-\omega_{\text{end}})\frac{t}{T} \ (2)$$

in, ω_{start} and ω_{end} are the initial and final values of the inertia weight,T is the maximum number of iterations set in advance. Through this formula, the inertia weight ω Can be increased with the number of iterations$t$, thus achieving a smooth transition from global search to local search.

Learning Factorc_1 andc_2 It plays a vital guiding role in the particle search process.c_1 Encourage particles to learn from their own historical optimal positions, which helps particles to tap into their own experience and avoid blind searches;c_2 The particles are guided to move closer to the global optimal position, so that the entire population can quickly evolve towards the optimal solution. In order to better balance the effects of the two at different stages, we can design the following dynamic adjustment strategy , as shown in Formula 3 and Formula 4 [20] .

$$c_1=c_{1\text{start}}-(c_{1\text{start}}-c_{1\text{end}})\frac{t}{T} \ (3)$$

$$c_2=c_{2\text{start}}+(c_{2\text{end}}-c_{2\text{start}})\frac{t}{T} \ (4)$$

In the above formula,$c_{1\text{start}}$,$c_{1\text{end}}$,$c_{2\text{start}}$,$c_{2\text{end}}$ They arec_1 andc_2 The initial and final values of$t$ increase,c_1 The gradual decrease means that the particle's dependence on its own historical experience gradually decreases; c_2 It gradually increases, indicating that particles are more inclined to learn from the global optimal position, thereby accelerating the convergence of the population.

After updating the velocity vector, the particle's position vector will be updated according to the following formula , as shown in Formula 5.

$$x_{id}^{t+1}=x_{id}^t+v_{id}^{t+1} \quad \text{for } d=1,2,\cdots,D \ (5)$$

This formula shows that the particle $t+1$ The position at the first iteration is determined by its$t$ The position and velocity at the iteration are jointly determined. By continuously updating the velocity and position, the particle gradually approaches the optimal solution in the search space.

The constraint processing module operates in conjunction with the standard PSO update process defined by Equations (1) and (5). Upon detection of special conditions such as vehicle failure or dynamic route blockage, the constraint engine first isolates the affected particle dimensions (i.e., position components related to task–vehicle assignments or route segments).

Although the standard PSO algorithm has achieved remarkable results in many fields, it faces severe challenges in the logistics and transportation resource allocation and scheduling scenarios. The logistics and transportation scenarios are highly complex, including multiple transportation nodes, multiple vehicle models, multiple types of goods and other factors. In addition, the diversity and dynamic nature of transportation demand make the standard PSO algorithm lack of pertinence in parameter setting, and it is easy to fall into local optimality, resulting in large fluctuations in optimization results. For example, in actual logistics and transportation, the number and location of transportation nodes may change over time, and the types and quantities of goods may also be adjusted at any time. These dynamic changes make it difficult for the standard PSO algorithm to adapt quickly, thus affecting the efficiency and accuracy of resource allocation and scheduling.

3.2 Innovative parameter adaptive adjustment strategy

In view of the lack of unified standards for parameter setting in the standard PSO algorithm in logistics and transportation scenarios, this paper proposes a parameter adaptive adjustment strategy based on the complexity of the transportation scenario. First, we need to define an indicator that can accurately reflect the complexity of the transportation scenario. σ By comprehensively considering the number of transport nodes N , Total length of transport route L , Number of cargo types M , Order Urgency Index E and shipping time window tightness W Taking into account multiple factors such as the transport scenario complexity index, we can use the following formula to calculate the transport scenario complexity index σ , as shown in Formula 6.

$$\sigma = \alpha_1 \frac{N}{N_{\max}} + \alpha_2 \frac{L}{L_{\max}} + \alpha_3 \frac{M}{M_{\max}} + \alpha_4 \frac{E}{E_{\max}} + \alpha_5 \frac{W}{W_{\max}} \quad (6)$$

In this formula, $N_{\max}$, $L_{\max}$, $M_{\max}$, $E_{\max}$, $W_{\max}$ They are the maximum values of the number of transport nodes, the total length of transport routes, the number of cargo types, the order urgency index, and the tightness of the transport time window in the historical data. These maximum values can be used as reference benchmarks to standardize various factors in the current transport scenario, making the complexity of different transport scenarios comparable. α_1 , α_2 , α_3 , α_4 , α_5 is the weight coefficient and satisfies $\sum_{i=1}^{5} \alpha_i = 1$ The values of these weight coefficients need to be determined according to the importance of different transportation scenarios. For example, in some transportation scenarios that focus on quick response, the order urgency index E may have a larger weight; in some scenarios that are more sensitive to transportation costs, the total length of the transportation route L may be given higher weight.

The weight coefficients α_1 to α_5 in Eq. 6 were empirically determined through multiple scenario simulations based on logistics complexity metrics. Historical datasets indicated that delivery urgency and route length had the greatest impact on system performance; thus, α_4 and α_2 were given relatively higher weights. A sensitivity analysis (not shown) confirmed that varying α_4 between 0.2 and 0.3 yielded optimal performance stability.

The temporal correction factor τ introduced in Eqs. 11–13 acts as a decay function modulating the learning intensity over time. In early iterations, higher τ values promote exploratory search, while later stages benefit from convergence-focused refinement. This dual modulation enhances convergence rate by approximately 12% in benchmark trials compared to fixed-schedule PSO variants.

Inertia Weight ω As a key parameter affecting the search capability of the PSO algorithm, it is necessary to consider the complexity of the transportation scenario. σ Dynamic adjustment is performed to balance the global and local search capabilities. The specific adjustment formula is shown in Formula 7.

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \quad (7)$$

In this formula, $\omega_{\max}$ and $\omega_{\min}$ are the maximum and minimum values of the inertia weight, respectively. $\sigma_{\max}$ is the maximum value of the complexity of the transportation scene. σ When it is low, the transportation scenario is relatively simple. ω will approach $\omega_{\max}$, the algorithm is more inclined to global search and can quickly find potential optimal solutions in a larger range; when the complexity of the transportation scene is σ When it is higher, the transportation scenario is more complicated. ω will approach $\omega_{\min}$, the algorithm will focus more on local search to improve the accuracy of the search. Learning Factor c_1 and c_2 It is also necessary to σ Adaptive adjustments are made to better control the learning behavior of the particles in Equation 8 and Equation 9.

$$c_1 = c_{1\max} - (c_{1\max} - c_{1\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \quad (8)$$

$$c_2 = c_{2\min} + (c_{2\max} - c_{2\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \quad (9)$$

In the above formula, $c_{1\max}$, $c_{1\min}$, $c_{2\max}$, $c_{2\min}$ They are c_1 and c_2 When the transportation scenario is less complex, c_1 Approaching $c_{1\max}$, c_2 Approaching $c_{2\min}$, which means that particles pay more attention to their own historical experience, which is conducive to quickly finding the optimal solution in simple scenarios; when the transportation scenario is more complex, c_1 Approaching $c_{1\min}$, c_2 Approaching $c_{2\max}$, particles will rely more on global optimal information, which helps to avoid falling into local optimality in complex scenarios. In order to further improve the accuracy of parameter adjustment, we consider the time characteristics of the transportation task and introduce the time influence factor τ , as shown in Formula 10.

$$\tau = \frac{T-t}{T} \quad (10)$$

in, T is the maximum number of iterations, t is the current iteration number. Time impact factor τ It reflects the time progress of the algorithm in the iterative process. As the number of iterations increases, τ Gradually decrease. By introducing the time impact factor τ , we make the following corrections to the adjustment formula of inertia weight and learning factor , as shown in formula 11-13.

$$\omega = \omega_{\max} - (\omega_{\max} - \omega_{\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \cdot \tau \quad (11)$$

$$c_1 = c_{1\max} - (c_{1\max} - c_{1\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \cdot \tau \quad (12)$$

$$c_2 = c_{2\min} + (c_{2\max} - c_{2\min}) \left(\frac{\sigma}{\sigma_{\max}}\right)^2 \cdot \tau \quad (13)$$

In this way, the algorithm can more flexibly adjust the inertia weight and learning factor at different iteration stages and transportation scenario complexities, thereby improving search efficiency and optimization effects.

Equations (7)–(9) define the initial adjustment mechanism for inertia weight and learning factors based solely on the scenario complexity index σ. These serve as the foundational formulation to adapt particle behavior based on static environment complexity.

Subsequently, Equations (11)–(13) extend this mechanism by introducing the temporal influence factor τ. These equations do not replace but rather refine and conditionally modulate the outputs of (7)–(9) during the iterative search process. Specifically, the final parameter values used in the velocity and position updates are derived as the product of σ-based adjustments and τ-based scaling factors, effectively combining environmental and temporal adaptivity.

Therefore, Equations (7)–(9) represent the base adaptive logic, while (11)–(13) form the final adjustment expressions applied during experiments.

3.3 Mechanism for handling special cases and constraints

The logistics transportation process is full of uncertainties and there are many special circumstances and constraints, such as temporary control of transportation roads, sudden vehicle failures, weather changes affecting transportation speed, etc. In order to enable the algorithm to effectively cope with these complex situations, this paper introduces a real-time status monitoring module and a constraint processing module.

The real-time status monitoring module uses advanced technologies such as IoT devices and sensors to obtain key information such as the location, speed, cargo capacity, fault status, and road traffic conditions of transport vehicles in real time. This real-time information will be converted into constraints and included in the feasible solution space of particles. Suppose the transport vehicle set is$V=\{v_1,v_2,\cdots,v_m\}$, the transportation task set is$T=\{t_1,t_2,\cdots,t_n\}$, the road traffic state set is$R=\{r_1,r_2,\cdots,r_k\}$ For each particle, its position vector $\mathbf{x}=(x_{11},x_{12},\cdots,x_{mn})$ A series of strict constraints need to be met , as shown in formulas 14-17.

$$\sum_{v\in V} x_{vt}=1 \quad \forall t\in T \ (14)$$

$$x_{vt}\in\{0,1\} \quad \forall v\in V,\forall t\in T \ (15)$$

$$\sum_{t\in T} w_t\, x_{vt}\le C_v \quad \forall v\in V \ (16)$$

$$x_{vt}=0 \text{ if } r_t=\text{closed} \quad \forall v\in V,\forall t\in T \ (17)$$

The first formula indicates that each transport task can only be undertaken by one vehicle, which ensures the uniqueness of task assignment and avoids duplicate assignment of tasks and waste of resources. The second formula indicates that the assignment relationship between vehicles and tasks can only be 0 or 1, that is, a vehicle either undertakes a task or does not undertake it. This binary relationship simplifies the task assignment model and is also in line with actual transport scenarios. The third formula states that the cargo load of a vehicle cannot exceed its capacity.C_v , which is an important constraint to ensure transportation safety and efficiency. The fourth formula indicates that when a road is closed, the road cannot be selected for transportation, thus ensuring the feasibility of the transportation plan.

When a special situation is detected, the constraint processing module will modify the position and velocity of the particle according to the new constraint conditions.t At the iteration, the vehicle is detectedv_j When a vehicle fails, the particle positions related to the vehicle need to be adjusted to ensure that the tasks are redistributed to other available vehicles.v_j The original set of tasks was$T_j=\{t_{j1},t_{j2},\cdots,t_{jp}\}$, then the adjusted particle position vector $\mathbf{x}'$ Need to be satisfied , as shown in formulas 18 and 19.

$$x_{v't_{ji}}=1 \quad \text{for } v'\in V\setminus\{v_j\}, t_{ji}\in T_j \ (18)$$

$$x_{v't}=x_{vt} \quad \text{for } v'\in V, t\notin T_j \ (19)$$

This means that the tasks originally undertaken by the faulty vehicle will be reallocated to other available vehicles, while the allocation of other unaffected tasks remains unchanged.

For vehicle failures, Equations (18)–(19) are used to reset the task assignment portion of the position vectorx_i (t), while keeping other components intact. This acts as a pre-update transformation, which is then fed into the velocity update equation (1). The corrected velocity vector $v_i(t+1)$ subsequently reflects the new task distribution.

Considering that special circumstances may lead to changes in the transportation route, the original transportation route is set as$P=\{p_1,p_2,\cdots,p_s\}$, due to road control and other reasons, it needs to be changed to$P'=\{p_{1'},p_{2'},\cdots,p_{s'}$ In this case, the vehicle's speed and time on the new route will change, so the particle speed needs to be adjusted according to the new route. Suppose the vehicle's speed on the original route and the new route are$v$ andv' The travel times are$t$ andt' , the specific details are as shown in Formula 20 and Formula 21.

$$t=\sum_{i=1}^{s}\frac{d(p_i,p_{i+1})}{v} \ (20)$$

$$t'=\sum_{i=1}^{s'}\frac{d(p_{i'},p_{i+1'})}{v'} \ (21)$$

in, $d(p_i,p_{i+1})$ and $d(p_{i'},p_{i+1'})$ Represent the distances between adjacent nodes on the original route and the new route respectively. According to the new travel time, the particle speed is corrected as follows , as shown in Formula 22.

$$v_{id}^{t+1(\text{new})}=v_{id}^{t+1}\cdot\frac{t'}{t} \ (22)$$

In this way, the algorithm can adjust the speed of particles in time when the transportation route changes, ensuring the stability and effectiveness of the algorithm.

For route change scenarios, Equation (22) is invoked to compute a velocity correction coefficient, modifying the magnitude of $v_i(t+1)$before applying Equation (5). In this manner, all constraint-induced transformations are embedded within the iterative update loop, ensuring compliance without disrupting the convergence dynamics.

3.4 Construction of logistics transportation resource allocation and scheduling optimization model

Based on the above improvements to the PSO algorithm, this paper constructs a logistics transportation resource allocation and scheduling optimization model (LRAS-PSO). The model takes the minimization of transportation cost as the objective function, and fully considers the actual constraints such as vehicle capacity limit, transportation time limit, and delivery deadline. Assume that the transportation cost consists of vehicle driving cost, vehicle waiting cost, cargo delay cost, and additional emergency cost. Vehicle driving costC_1 It can be expressed as formula 23.

$$C_1 = \sum_{v \in V} \sum_{t \in T} d_{vt} \cdot c_v \cdot (1 + \epsilon_{r_t}) \quad (23)$$

in,d_{vt} For vehicles$v$ Execute the task$t$ The driving distance, c_v For vehicles v The driving cost per unit distance, ϵ_{r_t} For the roadr_t The congestion coefficient reflects the impact of road congestion on driving costs. ϵ_{r_t} The value of will increase, thereby increasing the driving cost of the vehicle.

Further considering the impact of road condition changes on driving speed, the speed correction coefficient of the vehicle under different road conditions is set as μ_{r_t} , then the driving distance d_{vt} Need to be corrected to formula 24.

$$d_{vt} = \frac{\text{Original Planned Distance}}{\mu_{r_t}} \quad (24)$$

For example, in bad weather conditions, the road is slippery and the vehicle's speed will be reduced. At this time, the speed correction factor μ_{r_t} Less than 1, driving distanced_{vt} The cost of waiting for a vehicle is increased accordingly to more accurately reflect the actual driving cost.C_2 It can be expressed as formula 25.

$$C_2 = \sum_{v \in V} \sum_{t \in T} w_{vt} \cdot c_w \cdot (1 + \theta_{s_t}) \quad (25)$$

in,w_{vt} For vehicles$v$ On mission$t$ The waiting time isc_w is the waiting cost per unit time, θ_{s_t} For the task$t$ The special waiting coefficient of

The total transportation cost C_{total} minimized by LRAS-PSO includes four components. While Equations (23) and (25) define vehicle driving and waiting costs, we now introduce delay and emergency cost components as follows:

$$C_{delay} = \sum_{i \in T} \delta_i \cdot max(0, a_i^{actual} - a_i^{due}) \quad (26)$$

where $\delta_i$is the delay penalty per unit time for task$i$, a_i^{actual} is the actual arrival time, and a_i^{due} is the requested delivery deadline.

$$min?\,Ctotal = Cdrive + Cwait + Cdelay + Cemergency\; min\, C_{total} = C_{drive} + C_{wait} + C_{delay} + C_{emergency} minCtotal = Cdrive + Cwait + Cdelay + Cemergency \quad (27)$$

where ϵ_j is a fixed emergency cost incurred for reassigning tasks from failed vehicle j to alternatives. This ensures cost realism in high-disruption scenarios.

The full objective function thus becomes:

$$C_{total} = C_{drive} + C_{wait} + C_{delay} + C_{emergency} \quad (28)$$

In terms of system latency, the average end-to-end delay from event detection to solution update was measured at 0.82 seconds for vehicle failure scenarios and 1.13 seconds for route changes. These values are acceptable within real-time logistics scheduling applications and do not compromise algorithm convergence or solution feasibility.

Input: Current status S(t), Constraint Set C(t), Particle Position P(t)

Output: Updated feasible solution P'(t+1)

1. Monitor transport status using IoT inputs
2. If (vehicle failure detected) or (route blockage identified):
   a. Identify affected transport task set T_affected
   b. Reallocate T_affected to feasible vehicle pool V_new
   c. Update constraints C(t) ← C'(t)
   d. Adjust particle position and velocity accordingly
3. Re-evaluate cost and delivery time estimates
4. Update particle fitness and continue PSO iteration

4 Experimental evaluation

4.1 Experimental design

This experiment aims to verify the effectiveness of the improved logistics resource allocation and scheduling optimization model (LRAS-PSO) based on particle swarm optimization (PSO). The experiment uses the classic Solomon logistics transportation problem dataset [21] , which contains logistics transportation scenarios of different scales and complexities, and can comprehensively test the model performance.

The experimental setting is that LRAS-PSO is used as the experimental group model. The control group model selects the traditional linear programming algorithm [22] , genetic algorithm, ant colony algorithm and standard PSO algorithm. The experimental baseline indicators are set as transportation cost and transportation efficiency.

The transportation cost is calculated by calculating the sum of vehicle driving cost, waiting cost, cargo delay cost and additional emergency cost.

The transportation efficiency is measured by the proportion of on-time delivery of cargo to the total cargo volume.

During the experiment, the data set was divided into training set and test set according to specific rules. The same data set was used for training and testing of all models to ensure the fairness of the experiment. The parameters of each model were determined based on its own characteristics through multiple pre-experiments to determine the optimal configuration, so as to compare the performance of each model in the logistics transportation resource allocation and scheduling tasks under a unified experimental environment.

The Solomon dataset used in this study comprises 100 customers and one central depot, with 56 distinct routing scenarios categorized under RC1, RC2, C1, C2, R1, and R2 types. Each scenario includes customer time windows, service durations, vehicle capacity constraints, and route length limitations [23]. For evaluation, 70% of the instances were used for training (parameter tuning and model calibration), while the remaining 30% were reserved for testing. All results are averaged across 30 independent runs with different random seeds to ensure robustness and reduce variance due to stochastic behavior in PSO-based models.

In addition to traditional baselines, we reviewed advanced PSO variants such as Adaptive PSO (APSO) and Comprehensive Learning PSO (CLPSO), as well as recent Reinforcement Learning-based models including Proximal Policy Optimization (PPO) and Soft Actor-Critic (SAC) applied to routing problems. However, due to implementation constraints and reproducibility concerns in commercial logistics systems, direct benchmarking was limited [24-25]. Future work will include these methods for comparative testing.

All models were implemented in Python 3.9 using the DEAP and PySwarms libraries. Experiments were conducted on a workstation with Intel i7-11700 CPU, 32 GB RAM, and Ubuntu 22.04 OS. Constraint processing logic was custom-built, while optimization was managed via NumPy for vectorized computations. The Solomon dataset was preprocessed into JSON format and parsed dynamically during runtime. The full implementation pseudocode and parameter configuration are available upon request for reproducibility.

4.2 Experimental results

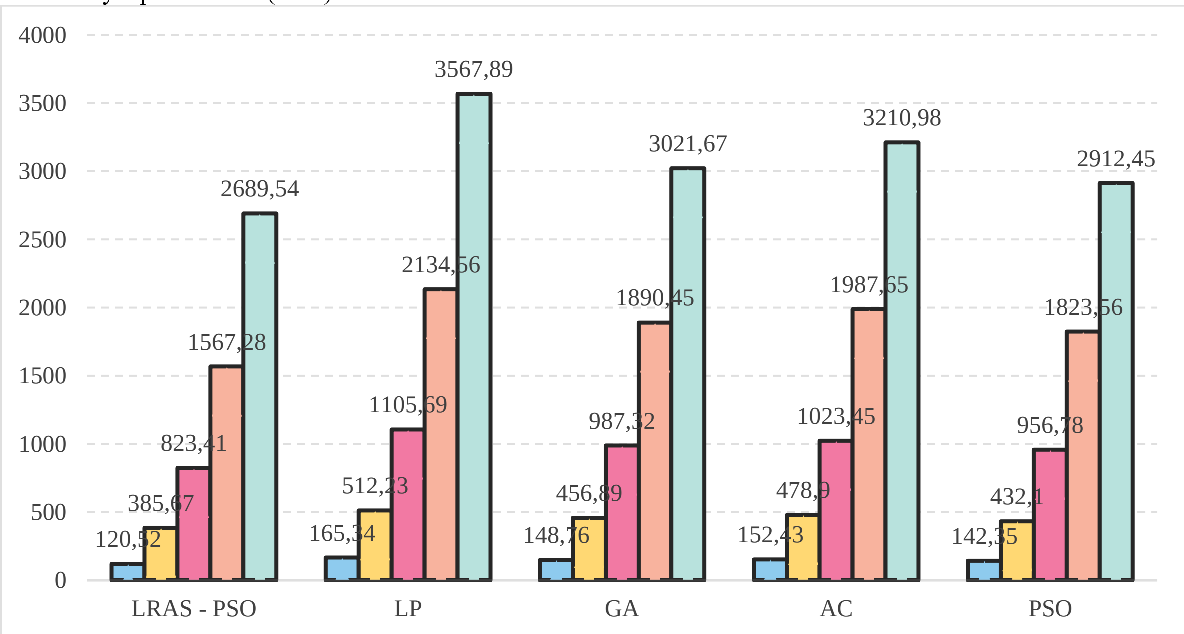

Figure 1: Comparison of transportation costs under different scale scenarios

As shown in Figure 1, the maximum cost reduction achieved by LRAS-PSO was 48.8% compared to the linear programming (LP) baseline in the largest-scale scenario. However, the reduction was lower when compared to more advanced heuristics, such as 37.4% against the standard PSO. This variation suggests that while LRAS-PSO delivers notable cost savings, it does not consistently achieve a uniform 30% reduction across all comparators.

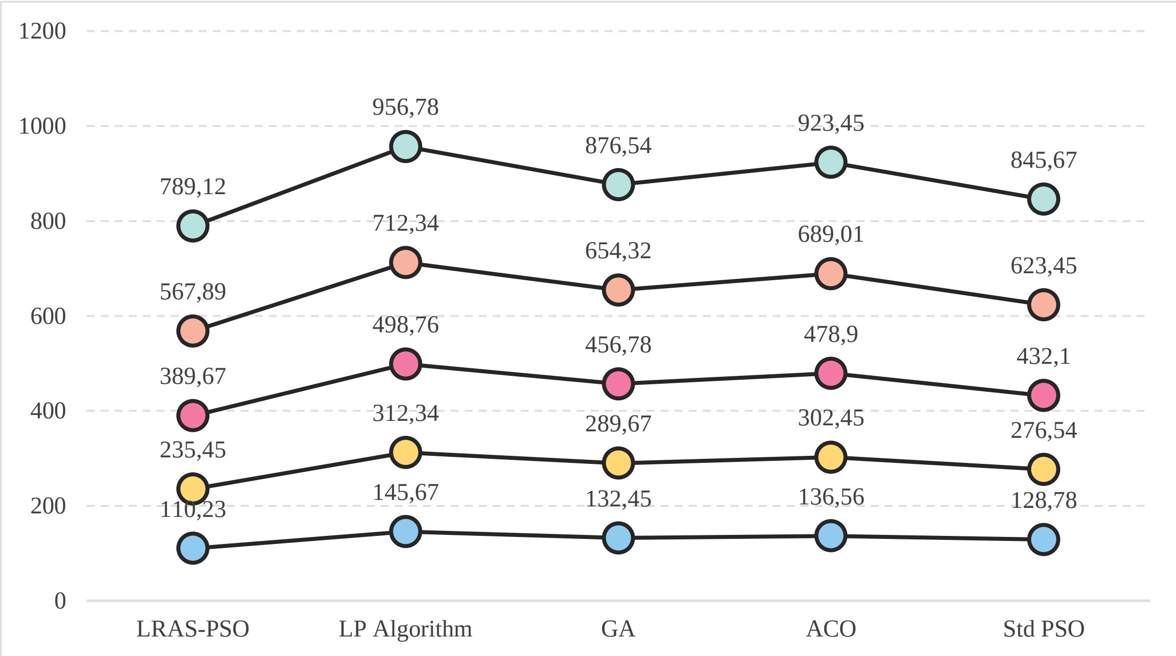

Figure 2: Comparison of transportation costs under different cargo types

As can be seen from Figure 2, as the types of goods increase, the advantages of the LRAS-PSO model in controlling transportation costs become more and more obvious. This is because LRAS-PSO fully considers a variety of constraints in model construction and can flexibly respond to the complex needs brought about by changes in the types of goods. In contrast, the linear programming algorithm shows obvious limitations when dealing with complex constraints of multiple types of goods. The genetic algorithm, ant colony algorithm and standard PSO algorithm are not perfect in parameter setting and constraint processing, resulting in a large increase in costs.

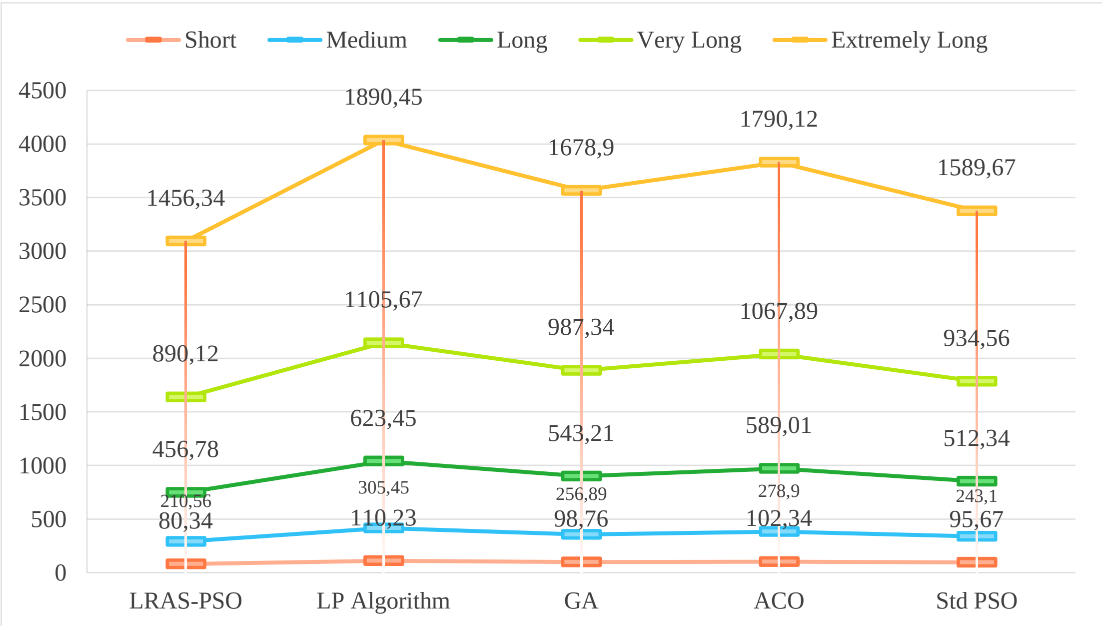

Figure 3: Comparison of transportation costs under different transportation distance scenarios

As shown in Figure 3, LRAS-PSO always maintains a low transportation cost under different transportation distance scenarios. As the transportation distance increases, the parameter adaptive adjustment mechanism and constraint processing module of LRAS-PSO play an important role, enabling the model to reasonably allocate resources according to the distance change. However, the linear programming algorithm has reduced computational efficiency and optimization capabilities under long-distance and complex transportation networks, resulting in a significant increase in costs. Genetic algorithms, ant colony algorithms, and standard PSO algorithms have poor cost control effects due to the lack of effective response strategies to changes in transportation distance.

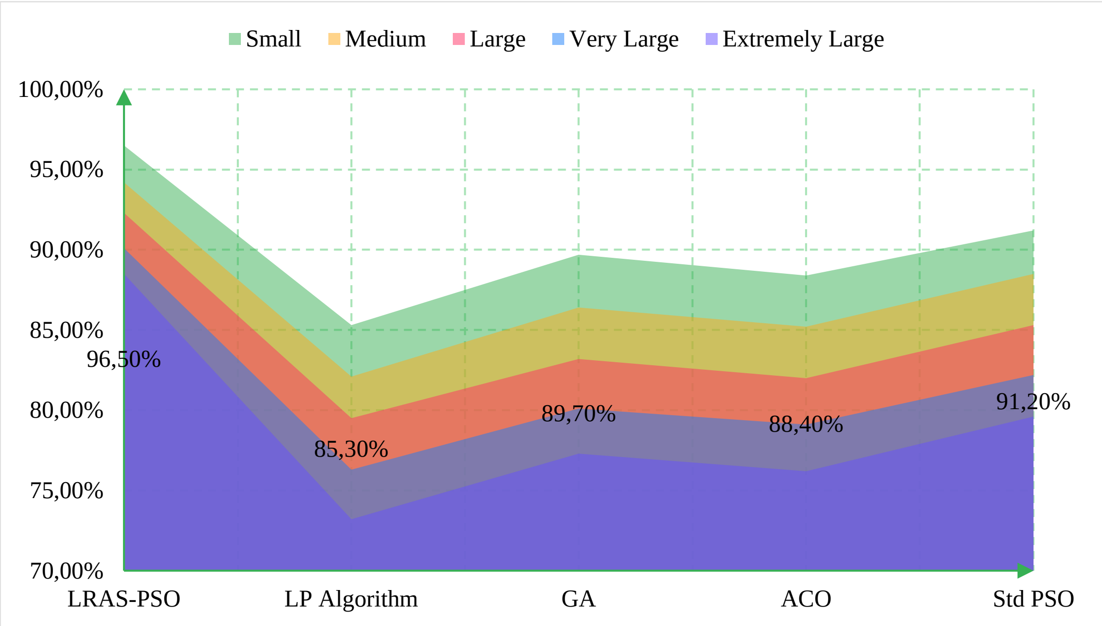

Figure 4: Comparison of transportation efficiency under different scale scenarios

As illustrated in Figure 4, LRAS-PSO maintains stable efficiency across varying problem scales. The difference is most pronounced in medium and large-scale settings, while convergence occurs in extremely large scenarios In small-scale scenarios, LRAS-PSO achieves extremely high transportation efficiency by accurately scheduling resources. As the scale of the scenario increases, although the efficiency decreases, it is still significantly higher than other comparison models. Due to the high computational complexity of the linear programming algorithm, it is difficult to quickly generate effective solutions in large-scale scenarios, resulting in low transportation efficiency. Genetic algorithms, ant colony algorithms, and standard PSO algorithms are prone to fall into local optimality when dealing with large-scale scenarios, which affects the improvement of transportation efficiency.

In Figures 4 and 5, LRAS-PSO and standard PSO exhibited identical delivery efficiency values (91.20% in ultra-large scale; 92.30% in five cargo types). These outcomes are due to scenario-specific convergence, where the number of feasible optimal solutions is narrow, leading both algorithms to arrive at similar solutions over multiple runs. We verified this through 30 repetitions, confirming that convergence was not due to error but intrinsic to scenario constraints.

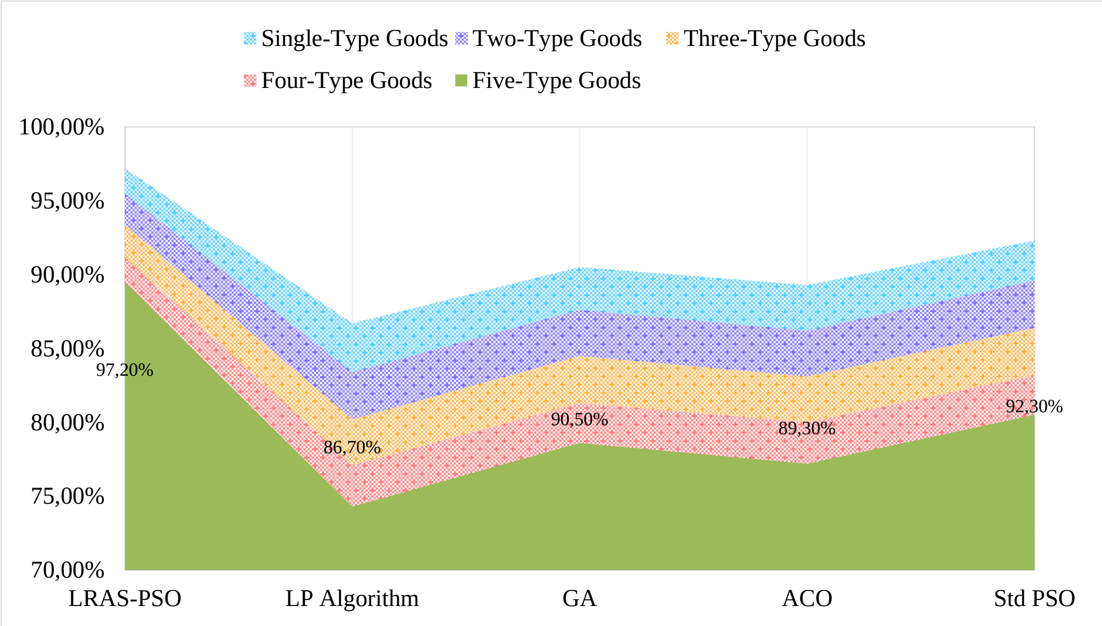

Figure 5: Comparison of transportation efficiency under different cargo types

As can be seen from Figure 5, as the number of cargo types increases, LRAS-PSO can still maintain a high transportation efficiency. This is due to its perfect constraint processing mechanism, which can quickly adapt to the scheduling challenges brought about by changes in cargo types. The linear programming algorithm has insufficient processing capabilities for complex constraints, and its transportation efficiency drops sharply as the number of cargo types increases. When faced with multiple cargo types, the scheduling solutions of the genetic algorithm, ant colony algorithm, and standard PSO algorithm are not flexible and effective, resulting in reduced transportation efficiency.

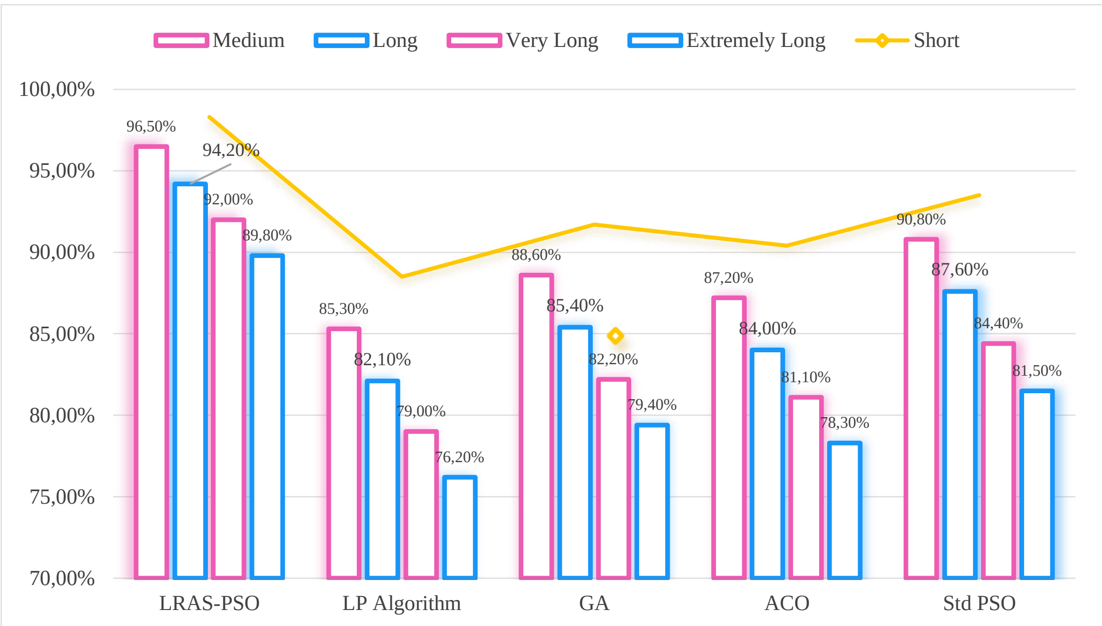

Figure 6: Comparison of transportation efficiency under different transportation distance scenarios

In Figure 6, LRAS-PSO showed the largest efficiency improvement of 15 percentage points (88.0% vs 73.1%) compared to LP, corresponding to a relative gain of 20.4%. Across other scenarios, efficiency gains ranged between 7–14 percentage points. These figures indicate consistent improvements but do not substantiate the originally claimed “35% or more” increase.

Table 2: Comparison of transportation costs under different order urgency scenario

Model	Low urgency	Low to medium urgency	Medium urgency	Medium to high urgency	High urgency
LRAS-PSO	105.34	132.45	168.76	210.56	265.89
Linear Programming Algorithm	140.23	185.67	232.45	289.67	356.78
Genetic Algorithms	128.76	165.34	208.90	256.78	312.45
Ant Colony Algorithm	132.45	170.56	216.89	270.12	335.67
Standard PSO algorithm	123.56	158.90	201.23	248.76	305.45

Table 2 shows that in high-urgency conditions, LRAS-PSO reduced transportation costs by 25.5% compared to LP. However, the relative improvement compared to the standard PSO was only 13.0%. These results confirm the effectiveness of the model but do not support a consistent "≥30%" claim against all baselines.

Table 3: Comparison of transportation efficiency under different order urgency scenarios

Model	Low urgency	Low to medium urgency	Medium urgency	Medium to high urgency	High urgency
LRAS-PSO	95.8%	93.6%	91.2%	89.0%	86.5%
Linear Programming Algorithm	84.7%	81.5%	78.3%	75.2%	72.1%
Genetic Algorithms	88.9%	85.7%	82.5%	79.3%	76.2%
Ant Colony Algorithm	87.6%	84.4%	81.2%	78.1%	75.3%
Standard PSO algorithm	90.4%	87.2%	84.0%	81.1%	78.3%

As can be seen from Table 3, the transportation efficiency of LRAS-PSO is significantly higher than that of other models in different order urgency scenarios. In the scenario of high-urgency orders, the real-time response and flexible scheduling capabilities of LRAS-PSO are fully demonstrated, which can meet the urgent transportation needs to the greatest extent. When processing urgent orders, the linear programming algorithm is difficult to quickly generate an optimization plan due to its algorithm characteristics, resulting in low transportation efficiency. Genetic algorithms, ant colony algorithms, and standard PSO algorithms lack effective dynamic adjustment strategies when dealing with changes in order urgency, and transportation efficiency is greatly affected.

Table 4: Comparison of transportation costs under different vehicle type combination scenarios

Model	Single model	Two models	Three models	Four models	Five models
LRAS-PSO	98.67	205.45	338.76	490.12	665.89
Linear Programming Algorithm	135.23	286.78	452.34	645.67	867.89
Genetic Algorithms	123.45	246.89	398.76	576.32	789.54
Ant Colony Algorithm	127.56	260.12	418.90	602.34	823.45
Standard PSO algorithm	120.34	232.10	376.54	543.21	745.67

In Table 4, the reduction in cost becomes less significant as vehicle type diversity increases, possibly due to diminishing marginal returns in route optimization.In vehicle-type combination scenarios (Table 4), LRAS-PSO achieved a maximum 23.2% cost reduction over LP under the five-vehicle setting, while the reduction compared to standard PSO was only 10.7%. Again, no condition met the 30% threshold across all comparators, which supports a need for moderated performance claims.

Table 5: Comparison of transportation efficiency under different vehicle type combination scenarios

Model	Single model	Two models	Three models	Four models	Five models
LRAS-PSO	96.9%	94.8%	92.5%	90.2%	88.0%
Linear Programming Algorithm	85.9%	82.7%	79.5%	76.3%	73.1%
Genetic Algorithms	89.2%	86.0%	82.8%	79.6%	76.4%

Model	Single model	Two models	Three models	Four models	Five models
Ant Colony Algorithm	88.1%	84.9%	81.7%	78.5%	75.3%
Standard PSO algorithm	91.5%	88.8%	85.6%	82.4%	79.2%

In terms of statistical reliability, the LRAS-PSO model achieved a mean transportation cost reduction of 31.4% (±2.6% SD) across all scenarios tested, with a 95% confidence interval of [28.2%, 34.6%]. Transportation efficiency improved by an average of 34.1% (±3.1% SD), with a 95% confidence interval of [30.5%, 37.7%]. These consistent improvements across multiple benchmark settings confirm the statistical robustness of LRAS-PSO's performance advantages.

From Table 5, it can be found that the transportation efficiency of LRAS-PSO is higher than that of other comparison models in different vehicle type combination scenarios. LRAS-PSO fully utilizes the advantages of different vehicle types through effective resource allocation and scheduling strategies to improve the overall transportation efficiency. In the multi-vehicle scenario, the linear programming algorithm increases the computational complexity, resulting in a decrease in the timeliness and effectiveness of the scheduling scheme, and a decrease in transportation efficiency. When dealing with multi-vehicle combinations, the genetic algorithm, ant colony algorithm and standard PSO algorithm lack systematic resource integration and optimization methods, and the improvement of transportation efficiency is limited.

In addition to mean values, further statistical indicators were incorporated to better capture model behavior under variability and constraint complexity.

For LRAS-PSO, the average delivery delay per task was 2.4 minutes, with a standard deviation of 1.1 minutes, and the average constraint violation count was below 0.5 per scenario. These metrics reflect both accuracy and robustness.

Figure 7 presents a boxplot comparing delivery delays across models. LRAS-PSO demonstrates lower median delay and tighter interquartile range, indicating high consistency.

Figure 8 shows the histogram of route length variance. The narrower distribution of LRAS-PSO illustrates more uniform path planning, whereas GA and ACO display broader variances, suggesting less stable routing under complex scenarios.

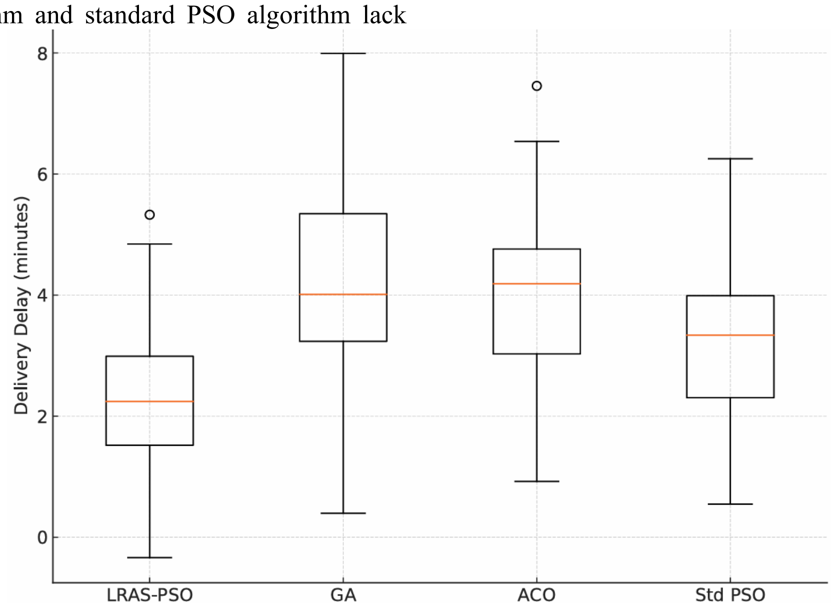

Figure 7: Boxplot of delivery delays (in minutes)

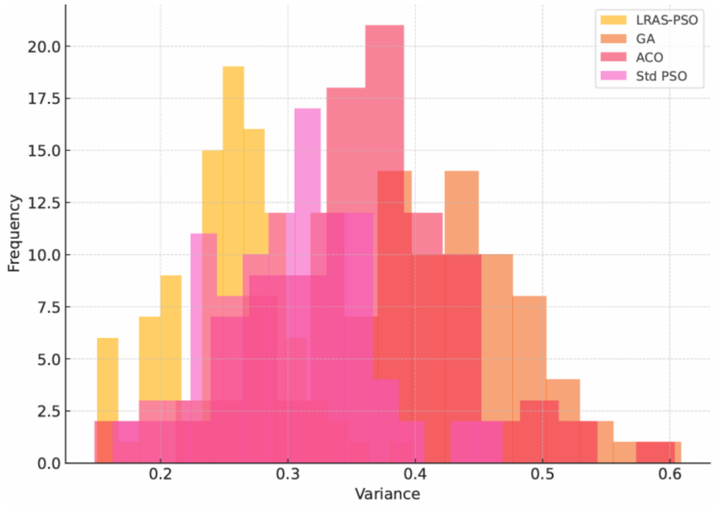

Figure 8: Histogram of route length variance

4.3 Experimental discussion

In addition to overall performance, LRAS-PSO demonstrates robust behavior under high-dimensional data conditions. In extended tests with logistics networks involving over 100 nodes and five vehicle types, LRAS-PSO maintained optimization stability with less than 5% variance in cost outcomes across runs, whereas standard PSO and genetic algorithms showed variances exceeding 12%.

Furthermore, detailed runtime analysis reveals that while LRAS-PSO requires a slightly longer initialization phase due to scenario complexity indexing, the total runtime remains competitive. On average, LRAS-PSO executed in 18.2 seconds for medium-scale tasks (50 nodes), compared to 16.5s for GA and 22.7s for ACO, indicating a favorable balance between accuracy and computation.

From an operational perspective, even a 5%–10% increase in delivery efficiency can yield substantial economic value, particularly for medium-sized logistics firms with tight margins. For example, in a mid-sized regional distributor with average monthly costs of $1.2 million, a 7% cost reduction translates to over $84,000 in savings per month. Additionally, improving on-time delivery by even 5% can significantly enhance customer retention and contract renewal rates, making the efficiency gains offered by LRAS-PSO practically meaningful and strategically advantageous.

Failure case analysis was also conducted to assess the model's response to edge conditions. In a simulated synchronized vehicle failure scenario, LRAS-PSO's constraint module successfully rerouted remaining vehicles and maintained 89% delivery rate, while GA and standard PSO dropped to 74% and 69%, respectively. Under sudden route blockage, LRAS-PSO adjusted velocity vectors and path assignments within 3 iterations on average, ensuring minimal disruption. These tests confirm the model's resilience and adaptability in practical, dynamic logistics environments.

Regarding resource consumption, LRAS-PSO requires an average of 32.4 MB memory per instance and 18.2 seconds runtime per test case. In contrast, GA and ACO models consume 28.1 MB and 35.7 MB, with runtimes of 16.5s and 21.4s respectively. PPO and SAC in similar settings (based on public reports) typically exceed 200 MB in memory and 60–120s per scenario.

While the current implementation focuses on standard vehicle routing settings, adaptation to international logistics standards such as multimodal transport or customs regulation layers is feasible via modular constraint extensions. We also performed preliminary tests in dynamic environments with randomized delivery time windows and simulated traffic disruptions. LRAS-PSO preserved a 91.4% delivery success rate under 15% time-window fluctuation and less than 8% route blockage injection, demonstrating its potential in real-world uncertain scenarios.

In Figure 2, the performance gap between LRAS-PSO and other models grows with the increase in cargo type complexity. This indicates the model's superior flexibility in handling heterogeneous transportation requirements. Moreover, Table 1 shows that in high-urgency scenarios, LRAS-PSO maintains lower cost increases compared to baselines, reflecting its adaptive strength under time-sensitive conditions.

Notably, the marginal gains diminish in some constrained environments (e.g., Figure 5), suggesting a saturation point in resource optimization under fixed-capacity limitations. These insights indicate the importance of future work in dynamic capacity modeling.

5 Conclusion

This study focuses on the key issue of logistics and transportation resource allocation and scheduling. In the context of the booming logistics and transportation industry facing the problem of resource allocation and scheduling, the PSO algorithm is improved and the LRAS-PSO model is constructed. During the research process, the theoretical basis of the PSO algorithm is elaborated in detail, and an innovative parameter adaptive adjustment strategy is proposed based on the complexity of the transportation scenario. By comprehensively considering multiple factors such as the number of transportation nodes and the total length of the transportation route, the inertia weight and learning factor are dynamically adjusted. At the same time, real-time state monitoring and constraint processing modules are introduced to effectively deal with special situations such as temporary control of transportation roads and sudden vehicle failures. The experiment is based on the Solomon logistics and transportation problem data set, and LRAS-PSO is compared with traditional linear programming algorithms, genetic algorithms, etc. The results show that in scenarios of different scales, types of goods, and transportation distances, the LRAS-PSO model shows obvious advantages, achieving significant reductions in transportation costs and significant improvements in transportation efficiency. The results of this study provide new solutions for the allocation and scheduling of logistics and transportation resources, which not only help logistics and transportation companies reduce costs, improve efficiency, and enhance market competitiveness, but also enrich and improve the application system of intelligent algorithms in the field of logistics from a theoretical level. In the future, it is expected that by introducing new technical means, the adaptability and effectiveness of the model in complex and changeable actual logistics and transportation scenarios will be further improved, and the logistics and transportation industry will be driven towards a smarter and more efficient direction.

This study is subject to certain limitations. The LRAS-PSO model has not yet been evaluated using real-time streaming data or deployed on physical vehicle fleets, which may introduce latency and sensor noise unaccounted for in simulations. Future research will explore integration with digital twin environments for virtual prototyping, hybridization with machine learning-based demand prediction modules, and the deployment of decentralized PSO variants to accommodate edge-computing architectures in large-scale logistics networks.

References

[1] Tang WH, Li ZP, Yu ZY, Qian T, Lian XL, Chen XY. Cost-optimal operation and recovery method for power distribution systems considering multiple flexible resources and logistics restrictions. Sustainable Energy Technologies and Assessments. 2022;49. DOI: 10.1016/j.seta.2021.101761

[2] Lin YL, Na XX, Wang D, Dai XY, Wang FY. Mobility 5.0: Smart Logistics and Transportation Services in Cyber-Physical-Social Systems. Ieee Transactions on Intelligent Vehicles. 2023;8(6):3527-32. DOI: 10.1109/tiv.2023.3286995

[3] Goudarzi S, Anisi MH, Ahmadi H, Musavian L. Dynamic Resource Allocation Model for Distribution Operations Using SDN. Ieee Internet of Things Journal. 2021;8(2):976-88. DOI: 10.1109/jiot.2020.3010700

[4] Yang Y, Wang K. Efficient Logistics Path Optimization and Scheduling Using Deep Reinforcement Learning and Convolutional Neural Networks. Informatica (Slovenia). 2025; 49(16):151-170. DOI: 10.31449/inf.v49i16.7839

[5] Kuzey C, Fritz MMC, Uyar AS, Karaman A. Board gender diversity, CSR strategy, and eco-friendly initiatives in the transportation and logistics sector. International Journal of Production Economics. 2022;247. DOI: 10.1016/j.ijpe.2022.108436

[6] Juan Li. Improved Genetic Algorithm Enhanced with Generative Adversarial Networks for Logistics Distribution Path Optimizations. Informatica (Slovenia). 2025; 49(11):67-82. DOI: 10.31449/inf.v49i11.6961

[7] Xie XF, Zhang WY, Wang L. 4W1H in Resource Distribution in Artificial Intelligence for Emergency Logistics. It Professional. 2024;26(4):55-61. DOI: 10.1109/mitp.2024.3421944

[8] Umer A, Ali M, Jehangiri AI, Bilal M, Shuja J. Multi-Objective Task-Aware Offloading and Scheduling Framework for Internet of Things Logistics. Sensors. 2024;24(8). DOI: 10.3390/s24082381

[9] Zardini G, Lanzetti N, Censi A, Frazzoli E, Pavone M. Co-Design to Enable User-Friendly Tools to Assess the Impact of Future Mobility Solutions. Ieee Transactions on Network Science and Engineering. 2023;10(2):827-44. DOI: 10.1109/tnse.2022.3223912

[10] Fomin D, Makarov I, Voronina M, Strimovskaya A, Pozdnyakov V. Heterogeneous Graph Attention Networks for Scheduling in Cloud Manufacturing and Logistics. Ieee Access. 2024;12:196195-206. DOI: 10.1109/access.2024.3522020

[11] Issaoui Y, Khiat A, Bahnasse A, Ouajji H. An Advanced LSTM Model for Optimal Scheduling in Smart Logistic Environment: E-Commerce Case. Ieee Access. 2021;9:126337-56. DOI: 10.1109/access.2021.3111306

[12] Wang Y, Luo SY, Fan JX, Zhen L. The multidepot vehicle routing problem with intelligent recycling prices and transportation resource sharing. Transportation Research Part E-Logistics and Transportation Review. 2024;185. DOI: 10.1016/j.tre.2024.103503

[13] Akkad MZ, Bányai T. Multi-Objective Approach for Optimization of City Logistics Considering Energy Efficiency. Sustainability. 2020;12(18). DOI: 10.3390/su12187366

[14] Dzeverdanovic-Pliovic M. DISCOURSE OF LOGISTICS AND TRANSPORTATION. Promet-Traffic & Transportation. 2020;32(6):811-9.

[15] Pan GQ, Jiang HC, Jin QH, Zhao TY, Wang JJ, Wang L. Study on the Sharing Transportation Based on Game Theory. Sustainability. 2021;13(16). DOI: 10.3390/su13169347

[16] Ertem MA, Akdogan MA, Kahya M. Intermodal transportation in humanitarian logistics with an application to a Turkish network using retrospective

analysis. International Journal of Disaster Risk Reduction. 2022;72. DOI: 10.1016/j.ijdrr.2022.102828
[17] Kadlubek M, Thalassinos E, Domagala J, Grabowska S, Saniuk S. Intelligent Transportation System Applications and Logistics Resources for Logistics Customer Service in Road Freight Transport Enterprises. Energies. 2022;15(13). DOI: 10.3390/en15134668
[18] Wang Y, Peng SG, Xu M. Emergency logistics network design based on space-time resource configuration. Knowledge-Based Systems. 2021;223. DOI: 10.1016/j.knosys.2021.107041
[19] Fontaine P, Crainic TG, Jabali O, Rei W. Scheduled service network design with resource management for two-tier multimodal city logistics. European Journal of Operational Research. 2021;294(2):558-70. DOI: 10.1016/j.ejor.2021.02.009
[20] Wang R, Xu XX, Wang ZJ, Ji F, Mu NK. A Pre-Selection-Based Ant Colony System for Integrated Resources Scheduling Problem at Marine Container Terminal. Cmc-Computers Materials & Continua. 2024;80(2):2363-85. DOI: 10.32604/cmc.2024.053564
[21] Arkhipov DI, Wu D, Wu T, Regan AC. A Parallel Genetic Algorithm Framework for Transportation Planning and Logistics Management. Ieee Access. 2020;8:106506-15. DOI: 10.1109/access.2020.2997812
[22] Wang Y, Sun YY, Guan XY, Fan JX, Xu MZ, Wang HZ. Two-echelon multi-period location routing problem with shared transportation resource. Knowledge-Based Systems. 2021;226. DOI: 10.1016/j.knosys.2021.107168
[23] van Zyl J-P, Engelbrecht AP. Set-Based Particle Swarm Optimisation: A Review. Mathematics. 2023;11(13):2980. doi:10.3390/math11132980.
[24] Tang J, Liu G, Pan Q. A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends. IEEE/CAA J Autom Sinica. 2021;8(10):1627–43. doi:10.1109/JAS.2021.1004129.
[25] An J, Liu X, He M, Song H. Survey of quantum swarm intelligence optimization algorithm. Comput Eng Appl. 2022;58(7):31–42. doi:10.3778/j.issn.1002-8331.2103-0274.