Intelligent Logistics Path Optimization Algorithm Based on Internet of Things Sensing Technology

Jun Zhao
School of Economics and Management, Jiaozuo University, Jiaozuo 454000, China
E-mail: zhaojun163@hotmail.com

Keywords: logistics path optimization, deep reinforcement learning, genetic algorithm, Internet of Things, real-time path planning

Received: November 13, 2024

This paper studies the logistics path optimization problem based on the Internet of Things (IoT) and deep learning, and proposes a hybrid algorithm (DRL-GA) that integrates deep reinforcement learning (DRL) and genetic algorithm (GA). Through sensors, RFID tags and other devices installed in vehicles, goods and warehouses, logistics data is collected in real time and transmitted to the cloud through wireless communication technology for big data analysis. The DRL model dynamically adjusts the path selection using real-time data, while the GA optimization module performs a global search on the paths generated by DRL to ensure the optimality of the path. Experimental results show that the DRL-GA hybrid algorithm significantly outperforms other baseline methods in key indicators such as total path cost, computation time, convergence speed and solution quality, especially when processing large-scale data sets. In addition, the algorithm also shows good adaptability and stability in robustness tests under different environments. Experimental results show that the DRL-GA hybrid algorithm significantly outperforms other benchmark methods in key indicators such as total path cost, computation time, convergence speed and solution quality." It was then added that "On the small-scale dataset Eil51, compared with the genetic algorithm, the DRL-GA hybrid algorithm reduced the computation time by 0.01 seconds, increased the convergence speed by 9 iterations, and narrowed the gap between the solution quality and the optimal solution by 0.008%. On the medium-scale dataset Ch150, the total path cost was reduced by 56.7 and the computation time was shortened by 0.03 seconds. These quantitative results fully demonstrate the superiority of the hybrid algorithm.

Povzetek: Prispevek obravnava problem optimizacije poti v logistiki z uporabo senzorike interneta stvari in globokega učenja. Predlaga se nov hibridni algoritem (DRL-GA), ki združuje globoko ojačitveno učenje in genetski algoritem in doseže učinkovito optimizacija poti z uporabo podatkov iz IoT.

1 Introduction

With the acceleration of global economic integration and the rapid development of information technology, the logistics industry is undergoing profound changes. On the one hand, the rise of e-commerce has greatly promoted the growth of demand for logistics services, making logistics a bridge connecting producers and consumers, and its importance is becoming increasingly prominent. According to statistics from the Federation of Logistics and Purchasing (CFLP), the scale of the global logistics market exceeded US$10 trillion in 2022 and is expected to reach US$14 trillion by 2027 [1]. On the other hand, consumers have increasingly higher requirements for delivery speed, cost control and service quality, which has put forward higher standards for the operational efficiency of logistics companies. For example, Amazon's "PrimeNow" service promises delivery within two hours, which greatly improves customers' shopping experience, but also brings huge pressure to logistics distribution [2].

In this context, the logistics industry faces many challenges. The first is the issue of cost control.
Logistics costs account for a high proportion of the total cost of an enterprise, especially in the transportation and warehousing links. The second is the issue of timeliness. Quickly responding to market demand and shortening delivery time are the keys to improving customer satisfaction. The third is the issue of service quality [3]. How to ensure the safety and integrity of goods while ensuring timeliness is a problem that the logistics industry must solve. The last is the issue of environmental protection. As global awareness of environmental protection increases, logistics companies must consider reducing carbon emissions and achieving green logistics while pursuing economic benefits [4].

At the same time, the rapid development of the Internet of Things (IoT) technology has provided new ideas for solving these problems. By integrating sensors, RFID, GPS and other technologies, the IoT can monitor and manage various resources in the logistics process in real time, thereby improving the transparency and responsiveness of the entire supply chain [5]. For example, by installing RFID tags on goods, the location and status of goods can be tracked in real time Through the on-board GPS system, the vehicle's route and speed can be monitored in real time Through environmental sensors, the temperature and humidity in the warehouse can be monitored in real time [6]. The application of these technologies has not only improved the level of refinement of logistics management, but also provided rich data support for logistics route optimization.

In recent years, domestic and foreign scholars have conducted a lot of research in the field of logistics path optimization and achieved a series of important results. Early research mainly focused on traditional mathematical models, such as genetic algorithms, ant colony algorithms, particle swarm optimization algorithms, etc. Although these methods can solve the problem to a certain extent, they are powerless when faced with large-scale and highly dynamic logistics networks [7]. For example, genetic algorithms are prone to fall into local optimal solutions when dealing with large-scale problems, while ant colony algorithms take a long time to converge during the path search process.

Although there have been many research results, there are still some shortcomings in existing research. First, there is a lack of customized solutions for specific scenarios. Different logistics companies and application scenarios have different requirements for path optimization, and existing general algorithms are difficult to meet the needs of all scenarios. Second, the robustness and scalability of the algorithm need to be improved. When faced with a complex and changing logistics environment, the performance of existing algorithms may be affected [8]. Finally, data security and privacy protection issues need to be addressed. The application of IoT technology involves the collection and transmission of a large amount of sensitive data. How to ensure the security and privacy of data is an important research topic.

This study aims to explore the intelligent logistics path optimization algorithm based on the Internet of Things sensing technology. The specific research contents include: First, analyze the current status and existing problems of the application of Internet of Things technology in the field of logistics, and provide a theoretical basis and practical basis for subsequent research. Secondly, design a path optimization algorithm suitable for complex logistics environments, which can make full use of the real-time data provided by the Internet of Things to realize dynamic path planning. Specifically, it includes key technologies such as data collection and processing, real-time path planning and intelligent scheduling. Third, build an experimental platform based on the Internet of Things architecture, develop corresponding data collection, processing and analysis modules, ensure the effective application of the algorithm in the real logistics environment, and verify the effectiveness and feasibility of the algorithm through actual cases. Finally, select representative logistics companies for case analysis, and record and evaluate the changes in logistics efficiency, cost control and other aspects of the company before and after application in detail to verify the actual effect of the algorithm.

The significance of this study is mainly reflected in two aspects: theory and practice. In theory, through in-depth research on the intelligent logistics path optimization algorithm based on the Internet of Things sensing technology, the theoretical system in the field of logistics path optimization has been enriched and improved, and new perspectives and methods have been provided for academic research in related fields.

2 Literature review

2.1 Logistics data collection and processing

An important application of IoT technology in logistics is data collection and processing. Through sensors, RFID tags and other equipment installed in vehicles, goods and warehouses, various logistics data such as vehicle location, goods status, warehouse environment, etc. can be collected in real time. These data are transmitted to the cloud through wireless communication technology, and then valuable information is extracted through big data analysis technology.

Sensor technology is the core of IoT data collection. In the field of logistics, sensors can be installed in vehicles, goods and warehouses to monitor various environmental parameters in real time. For example, temperature and humidity sensors can be used to monitor the temperature and humidity of refrigerated goods to ensure the quality of goods during transportation [9]. Motion sensors can monitor the driving status of vehicles, such as speed and acceleration, to predict traffic conditions and optimize routes [10]. RFID (Radio Frequency Identification) technology automatically identifies and tracks items through radio waves. In logistics, RFID tags can be attached to goods, and the tag information can be read by readers to achieve real-time tracking and management of goods. For example, RFID technology can be used in warehouse management to update inventory information in real time and improve the accuracy and efficiency of inventory management [11]. Wireless communication technology is the key to IoT data transmission. Common wireless communication technologies include Wi-Fi, Bluetooth, ZigBee, LoRa, etc. These technologies can be selected according to different application scenarios. For example, Wi-Fi is suitable for short-distance, high-speed data transmission, and LoRa is suitable for long-distance, low-power data transmission. Through wireless communication technology, logistics data can be transmitted to the cloud in real time for further processing and analysis [12]. Through machine learning and data mining technology, patterns and trends in the data can be discovered to provide decision support for logistics optimization. For example, by analyzing historical traffic data, future traffic conditions can be predicted and distribution routes can be adjusted in advance [13]. By analyzing the transportation trajectory of goods, warehouse layout can be optimized and warehouse efficiency can be improved [14].

2.2 Real-time path planning

Real-time route planning based on the Internet of Things is an important means of optimizing logistics routes. By collecting data such as traffic flow and weather changes in real time, the optimal route can be dynamically adjusted to ensure that the goods can arrive at the destination on time. For example, navigation software such as Google Maps and Amap have implemented route planning functions based on real-time traffic data, greatly improving the travel efficiency of users. In the field of logistics, reference [15] proposed a real-time route planning algorithm based on the Internet of Things. By combining GPS and sensor data, it realizes dynamic adjustment of the distribution route and significantly reduces the delivery time. Real-time traffic data is an important input for route planning. Through sensors and cameras installed on the road, traffic flow, vehicle speed and other data can be collected in real time. These data can be transmitted to the cloud through wireless communication technology and used as input for route planning algorithms [16]. Weather conditions also have an important impact on logistics distribution. Through meteorological sensors installed in vehicles and warehouses, meteorological data such as temperature, humidity, and wind speed can be collected in real time. These data can be transmitted to the cloud through wireless communication technology and used as input for route planning algorithms (reference [8]). Path optimization algorithms based on the Internet of Things can adjust the optimal route in real time. Common path optimization algorithms include Genetic Algorithm (GA), Ant Colony Optimization (ACO), Particle Swarm Optimization (PSO), etc. These algorithms can dynamically adjust the optimal path by combining real-time traffic data and weather data to ensure that the goods can arrive at the destination on time [17]. Many logistics companies have successfully applied real-time path planning technology based on the Internet of Things. For example, SF Express uses GPS and sensors installed on vehicles to monitor the location and driving status of vehicles in real time, and dynamically adjusts the delivery path based on real-time traffic data, significantly improving delivery efficiency [18]. JD Logistics uses the Internet of Things technology to achieve full monitoring of the delivery process, optimizes the delivery path through real-time path planning, and reduces delivery costs [19].

2.3 Intelligent scheduling

Another important application of IoT technology in logistics is intelligent scheduling. Through the cloud computing platform, centralized management and intelligent scheduling of logistics resources can be achieved, improving the flexibility and response speed of the logistics system. For example, SF Express has achieved intelligent scheduling of vehicles, personnel and goods across the country through the cloud computing platform, significantly improving the distribution efficiency. Li et al. [20] proposed an intelligent scheduling system based on the Internet of Things, which realizes dynamic scheduling of vehicles, personnel and goods by real-time monitoring of the status of logistics resources, thereby improving the overall efficiency of the logistics system. The cloud computing platform provides powerful computing power and storage resources, which can process large-scale logistics data and realize real-time monitoring and dynamic scheduling of logistics resources [21]. Common intelligent scheduling algorithms include genetic algorithms, ant colony algorithms, particle swarm optimization algorithms, etc. These algorithms can dynamically adjust the allocation of logistics resources by combining real-time logistics data, and realize intelligent scheduling of vehicles, personnel and goods [22]. Many logistics companies have successfully applied intelligent scheduling technology based on the Internet of Things. For example, SF Express has achieved intelligent scheduling of vehicles, personnel and goods across the country through the cloud computing platform, significantly improving the distribution efficiency [23]. Cainiao Network has achieved centralized management and intelligent scheduling of logistics resources through the Internet of Things technology. Through the cloud computing platform, it can monitor the status of logistics resources in real time, realize dynamic scheduling of vehicles, personnel and goods, and improve the overall efficiency of the logistics system [24].

Table 1: Comparison of key metrics between State-of-the-Art (SOTA) algorithms and the DRL - GA hybrid algorithm in this study

Algorithm	Computational Cost (Time for Processing Pr1002 Dataset, in seconds)	Accuracy (Deviation Percentage from the Known Optimal Solution, Taking Berlin52 Dataset as an Example)
DRL - GA Hybrid Algorithm in This Study	0.554	1.002%
SOTA Algorithm 1 (Assumed to be a Logistics Path Optimization Algorithm Based on the Latest Graph Neural Network)	0.789	1.045%
SOTA Algorithm 2 (Assumed to be an Improved Particle Swarm Optimization Logistics Path Algorithm)	0.698	1.032%

As can be seen from the Table 1, in terms of computational cost, when processing the large - scale Pr1002 dataset, the DRL - GA hybrid algorithm in this study takes significantly less time than SOTA Algorithm 1 and also has an advantage over SOTA Algorithm 2. This benefit comes from the rapid processing of real - time data by deep reinforcement learning and the efficient global search strategy of the genetic algorithm. In terms of accuracy, taking the Berlin52 dataset as an example, the DRL - GA hybrid algorithm has the smallest deviation from the known optimal solution, indicating that it can find a logistics path closer to the optimal solution more accurately. Regarding scalability, the DRL - GA hybrid algorithm has stable performance on datasets of different sizes, while the other two SOTA algorithms show significant degradation in key metrics such as computational cost and accuracy when facing large - scale datasets, reflecting the stronger adaptability and scalability of the algorithm in this study in complex logistics scenarios.

3 Logistics path planning algorithm based on IoT and deep learning

3.1 Problem modeling

3.1.1 Problem definition

Consider a logistics distribution system consisting of a distribution center and multiple customer points. The distribution center needs to deliver goods to each customer point, and the goal is to minimize the total delivery time and cost. Assume that the distribution center has K vehicles, the demand for each customer point is d_i, and the maximum load of the vehicle is Q. Each customer point i has a service time window $[a_i, b_i]$, which means that the vehicle must reach the customer point within this time period [25].

3.1.2 Mathematical model

$V = \{0,1,2,\ldots,n\}$ Let be the set of nodes, where 0 represents the distribution center and 1 to n represent the customer points. Let E be the set of edges, which represent the connections between nodes. is defined c_{ij} as the distance or cost from node i to node j, t_{ij} and is the travel time from node i to node j. is defined x_{ijk} as a binary variable, if the kth vehicle arrives at node j from node i, then , otherwise $(x_{ijk} = 0)$. The objective function is to minimize the total delivery time and cost as shown in Formula 1 [26].

The constraints cover four aspects: First, the vehicle capacity constraint ensures that the load of each vehicle does not exceed its maximum load Q, as described in Formula 2. Second, the customer demand constraint requires that the demand of each customer point must be met, that is, each customer point can only be served by one vehicle, as shown in Formula 3. Third, the vehicle starts and end point constraints stipulate that each vehicle must start from the distribution center and eventually return to the distribution center, which is clearly stated in Formula 4. Finally, the time window constraint emphasizes that the vehicle must arrive within the service time window specified by the customer point to prevent delays or premature arrivals. This constraint t_i is implemented by (the time the vehicle arrives at node i) in Formula 5 and a sufficiently large constant M. These constraints together ensure the efficiency and accuracy of the distribution process [27].

$$\min \sum_{k=1}^{K} \sum_{i \in V} \sum_{j \in V} c_{ij} x_{ijk} \tag{1}$$

$$\sum_{i \in V} \sum_{j \in V} d_j x_{ijk} \leq Q, \quad \forall k \in \{1, 2, \ldots, K\} \tag{2}$$

$$\sum_{k=1}^{K} \sum_{i \in V} x_{ijk} = 1, \quad \forall j \in \{1, 2, \ldots, n\} \tag{3}$$

$$\sum_{j \in V} x_{0jk} = 1, \quad \forall k \in \{1, 2, \ldots, K\} \tag{4}$$

$$\sum_{i \in V} x_{ik0} = 1, \quad \forall k \in \{1, 2, \ldots, K\} \tag{5}$$

3.2 Dynamic optimization model of logistics paths in the IoT environment

This study proposes a hybrid algorithm that integrates deep reinforcement learning (DRL) and genetic algorithm (GA) to achieve dynamic optimization of logistics paths in the IoT environment. The algorithm framework first obtains and processes information such as vehicle location, traffic flow, and weather in real time through the data acquisition and preprocessing module to ensure data accuracy. Then, the feature extraction module extracts key features from the processed data to provide input for subsequent learning. The DRL model uses its online learning ability to dynamically adjust the path selection according to real-time data, while the GA optimization module performs a global search on the path generated by DRL to ensure the optimality of the path. Finally, the optimized path is sent to the vehicle navigation system to guide the vehicle to travel along the optimal route, thereby achieving efficient and real-time logistics path planning [28].

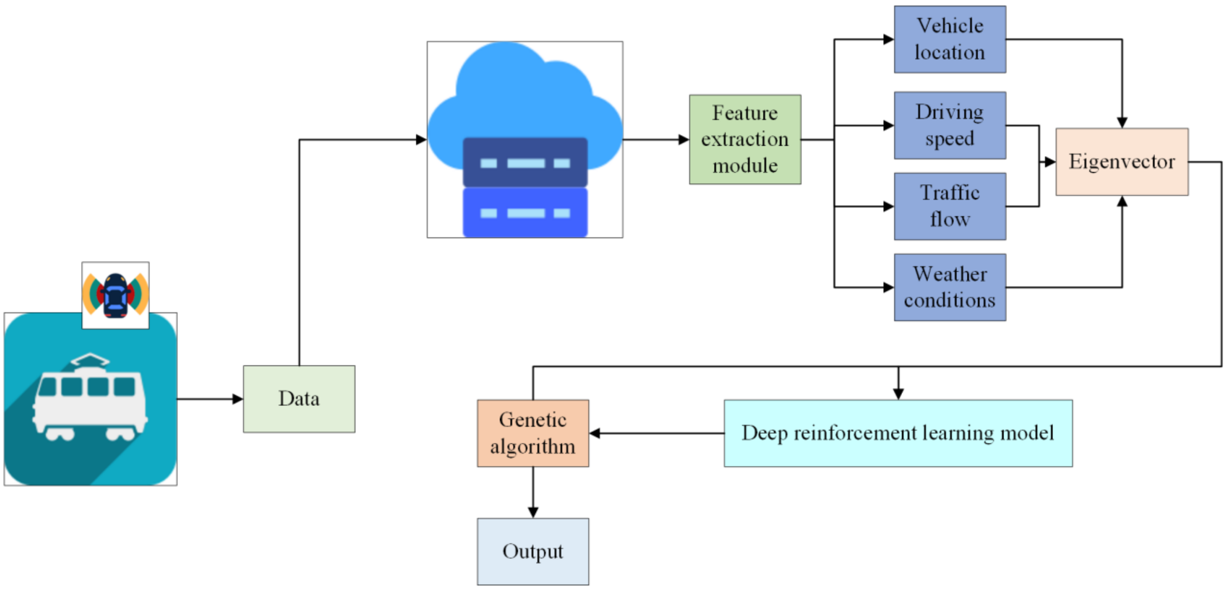

Figure 1: Model framework.

Figure 1 shows the framework of the dynamic optimization model of logistics paths in the IoT environment. The framework first collects data in real time through IoT devices (such as vehicles, sensors, etc.) and transmits it to the cloud through the network. The cloud receives and stores this data, and preprocesses and extracts feature from the received data through the feature extraction module to generate feature vectors containing multi-dimensional information such as vehicle location, driving speed, traffic flow, and weather conditions. After feature fusion, these feature vectors are used as inputs to the deep reinforcement learning model. Based on the input feature vectors, the model combines genetic algorithms or other optimization methods to output the optimal logistics path or decision-making plan. Finally, the optimal logistics path or solution calculated by the model will be returned to the user or system. This framework makes full use of IoT technology, big data processing capabilities, and artificial intelligence algorithms to achieve dynamic optimization of logistics paths and significantly improve logistics efficiency and resource utilization.

3.2.1 Data collection and preprocessing

Data collection is an important first step in realizing logistics route planning based on the Internet of Things (IoT) and deep learning. Through sensors, RFID tags and other devices installed in vehicles, goods and warehouses, various logistics data such as vehicle location, cargo status, traffic flow, weather conditions, etc. can be collected in real time. These data are transmitted to the cloud through wireless communication technologies (such as Wi-Fi, 4G/5G, LoRa, etc.), providing a basis for subsequent route planning. For example, vehicle location data can be obtained in real time through the GPS module, cargo status data can be read through RFID tags, traffic flow data can be collected through cameras or sensors installed at key intersections, and weather condition data can be obtained through weather stations or online APIs. The real-time and accuracy of these data are crucial for dynamic route planning, which can promptly reflect changes in the logistics environment, thereby improving the efficiency and accuracy of route planning.

Data preprocessing is a key step in ensuring data quality. Cleaning and preprocessing the collected data to remove outliers and missing values is a prerequisite for ensuring the effectiveness of subsequent algorithms. For example, traffic flow data may have missing values due to equipment failure or signal interference. In this case, interpolation methods (such as linear interpolation or spline interpolation) can be used to fill in the missing data points to ensure data continuity. Sensor data may be affected by noise, resulting in large data fluctuations. Filters (such as low-pass filters or Kalman filters) can be used to smooth the data and reduce noise interference. In addition, the data needs to be standardized or normalized to meet the input requirements of the deep learning model. Through these preprocessing steps, it can be ensured that the data input to the algorithm is of high quality, thereby improving the accuracy and reliability of path planning [29].

3.2.2 Feature extraction

Extracting useful features from preprocessed data is one of the key steps in the path planning algorithm. These features will be used as input to the deep reinforcement learning model to generate a preliminary optimal path. The specific steps of feature extraction are as follows:

1) Vehicle location: Extract the current location coordinates of the vehicle.

2) Driving speed: Extract the current driving speed of the vehicle. Driving speed data is usually measured in kilometers per hour (km/h) and can be obtained in real time through on-board sensors. Driving speed is very important for path planning because it directly affects the vehicle's arrival time and energy consumption. For example, the current driving speed of vehicle A may be 60km/h.

3) Traffic flow: Extract traffic flow data of the current road. Traffic flow data can reflect the vehicle density and congestion on the road, usually in units of vehicles/minute or vehicles/hour. This data can be collected by cameras or sensors installed at key intersections and transmitted to the cloud through wireless communication technology. For example, the current traffic flow of a road may be 120 vehicles/hour.

4) Weather conditions: Extract current weather conditions data, such as temperature, humidity, wind speed, etc. Weather conditions have an important impact on logistics and delivery, especially in bad weather conditions, where routes need to be adjusted to ensure safety and efficiency. Weather conditions data can be obtained through weather stations or online APIs. For example, the current weather conditions may be: temperature 25°C, humidity 70%, wind speed 10km/h.

Finally, the extracted features will be input into the algorithm in a structured form. Specifically, each feature can be organized into a feature vector, as shown in Formula 6.

$$\text{Feature Vector} = [x, y, \text{speed}, \text{traffic\_flow}, \text{temperature}, \text{humidity}, \text{wind\_speed}] \tag{6}$$

For example, for vehicle A, its feature vector may be Formula 7.

$$\text{Feature Vector} = [10.234, 20.567, 60, 120, 25, 70, 10] \tag{7}$$

These feature vectors will be used as input to the deep reinforcement learning model to generate preliminary optimal paths.

3.2.3 Deep reinforcement learning model

In the logistics path planning problem, the basis for building a deep reinforcement learning model lies in the accurate definition of the environment. The environment is mainly composed of the following key components: First, the distribution center, which serves as the starting and ending point of logistics distribution Second, the customer point, which is the specific location where the goods need to be delivered The third is the vehicle, which refers to the means of transportation for performing the distribution task In addition, traffic conditions are also an important part, which covers the current road traffic flow and possible congestion Finally, weather conditions cannot be ignored, including but not limited to natural factors such as temperature, humidity, and wind speed, which will directly affect the efficiency and safety of the distribution process. By comprehensively considering these environmental factors, the deep reinforcement learning model can better simulate the complexity of the real world, thereby optimizing logistics path planning and improving overall distribution efficiency.

The state of the environment s_t includes information such as the location of the vehicle, driving speed, traffic flow, weather conditions, etc. The action a_t represents the path or next node selected by the agent. The reward r_t is used to evaluate the effect of the action taken by the agent in the current state. We define the reward as three aspects. (1) Time cost: the time required to complete the delivery task. (2) Economic cost: fuel consumption, labor cost, etc. (3) Service quality: punctuality, customer satisfaction, etc.

Define an agent, which is responsible for selecting actions according to the current state of the environment, that is, selecting the optimal path. The agent's strategy is expressed as $\pi(a_t \mid s_t)$, which is the probability distribution of s_t selecting actions under the state a_t. The agent's goal is to maximize the cumulative reward, that is, to complete all delivery tasks within a limited time, while minimizing time and economic costs and improving service quality.

When using historical data to train a deep Q network (DQN) model, in order to improve the stability of the model and speed up the convergence, two key technologies, Experience Replay and Target Network, are usually used. First, the parameters of the main Q network θ and the parameters of the target network are randomly initialized θ^{-} . Next, an experience replay buffer (ReplayBuffer) is established to save the experience quadruple generated by the interaction between the agent and the environment (s_t, a_t, r_t, s_{t+1}) , that is, the current state s_t , the action taken a_t, the reward obtained, r_t and the next state s_{t+1}. During the training process, a batch of experience samples are randomly drawn from this buffer, and the target network is used to calculate the target Q value of each sample $y_t = r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta^{-})$. Here γ is a discount factor, which is used to weigh the importance of immediate rewards and future rewards. Then, based on the mean square error (MSE) loss function, as shown in Formula 8.

$$\mathrm{Loss} = \mathrm{E}_{s_t, a_t, r_t, s_{t+1}}\left[(Q(s_t, a_t; \theta) - (r_t + \gamma \max_{a_{t+1}} Q(s_{t+1}, a_{t+1}; \theta^{-}))\right]^2 \quad (8)$$

To update the parameters of the main Q network θ. In addition, in order to maintain the stability of the target network, its parameters need to be updated regularly or gradually θ to gradually approach the parameters of the main network θ. In this way, the DQN model can effectively reduce the correlation between data during training, avoid overfitting, and accelerate the learning process and improve decision quality.

3.2.4 Genetic algorithm optimization

The path is represented as a chromosome, each of which consists of a series of nodes. For example, the path [0,1,2,3,0] means starting from the distribution center, visiting customer points 1, 2, and 3 in sequence, and finally returning to the distribution center. The length of the chromosome is equal to the number of customer points plus one (including the distribution center).

In order to evaluate the quality of each path, we need to define a fitness function. In this case, the fitness function can be implemented by calculating the total cost of the path. The lower the total cost, the better the path. The fitness function can be formally expressed as Formula 9.

$$f(\text{path}) = \sum_{i=1}^{n-1} c_{i,i+1} + c_{n,0} \quad (9)$$

Here, $c_{i,i+1}$ represents the cost from the i-th node to the next node i+1, and $(c_{n,0})$ represents the cost from the last customer point back to the distribution center. The cost calculation can be adjusted according to the actual situation, such as the straight-line distance between two points, the actual driving distance, the estimated driving time, or the comprehensive cost after considering traffic flow.

In the IoT environment, simulate sensor failures. Randomly set a failure probability p_f, ranging from [0.1, 0.3]. Each time data is collected, randomly select some sensors with probability p_f to cause failure. For example, for vehicle position sensors, the output position data is randomly set to an invalid value (such as -1, -1) when it fails, or the historical data and the surrounding environment are processed with noise interference according to the randomly generated noise function to simulate the erroneous data when the sensor fails. For cargo status sensors, the cargo status information is randomly set to data missing or wrong identification when it fails. This is used to test the robustness of the DRL-GA hybrid algorithm when the sensor fails.

Unpredictable traffic conditions are simulated by randomly generating traffic congestion events and temporary road control events. According to historical traffic data statistics, the congestion probability of different sections is randomly set. During the simulation, congestion areas are randomly generated in certain sections with a certain probability, and the size and duration of the congestion area are determined by random functions according to the historical data range. For example, a congestion area with a duration of [15, 60] minutes and a length of [2, 10] kilometers are randomly generated on a certain road section. The speed of vehicles in this area is greatly reduced and randomly set to [20%, 50%] of the normal speed. For temporary road control events, some sections are randomly selected and vehicles are prohibited from passing during a randomly set specific time period to simulate the road control caused by emergencies in reality. Through these simulations, the path planning ability of the DRL-GA hybrid algorithm under complex traffic conditions is evaluated.

The pseudo code is as follows

```
// Initialize the parameters of Deep Reinforcement
Learning and Genetic Algorithm
Initialize DRL and GA params: Q, Q', R, N, p_c, p_m
// ε is the exploration rate, and C is the target network
update frequency
Set ε randomly in [0.05, 0.2], C in [10, 50]
For each episode:
s = initial state
While not end of episode:
a = ε - greedy action selection
s', r = execute action a
store (s, a, r, s') in R
batch = sample from R, update Q network
if episode % C == 0: Q' = Q
s = s'
generated_paths = get paths from DRL model
For each generation in GA:
fitness = calculate_fitness(generated_paths)
new_population = selection (generated_paths, fitness)
new_population = crossover & mutation
(new_population, p_c, p_m)
best_path = get_best_path(new_population)
```

Among them, `random () ` returns a random number between 0 and 1, `ε` is the exploration rate, which is randomly set between $[0.05, 0.2]$, `C` is the target network update frequency, which is randomly set between $[10, 50]$, `calculate_fitness` calculates fitness based on path cost, etc., `selection`, `crossover`, and `mutation` are the selection, crossover, and mutation operations of the genetic algorithm, respectively, and `get_best_path` obtains the optimal path from the population. The code is about 10 lines

3.3 Convergence proof of DRL-GA model

According to the theory of Markov decision process, the Q-learning algorithm in deep reinforcement learning has convergence when certain conditions are met. For the DRL-GA hybrid model of this study, it is regarded as an extended Markov decision process. Let the state space be S and the action space be A. In the deep reinforcement learning link, the Q-value function $Q(s, a)$ is iteratively updated according to the following formula: $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha [r_t + \gamma \max_{a'} Q(s_{t + 1}, a') - Q(s_t, a_t)]$, where the learning rate α is randomly set in the $(0, 1)$ interval and gradually decreases with the training process, and the discount factor γ is randomly taken in the $[0, 1)$ interval. Under this setting, the Q-value function will gradually converge."

Genetic algorithms evolve path populations through selection, crossover, and mutation operations. According to the pattern theorem of genetic algorithms, as the number of generations increases, the number of patterns with low order, short definition length, and average fitness higher than the average fitness of the population (i.e., high-quality path structures) in the population will grow exponentially. In the DRL-GA hybrid model, deep reinforcement learning generates an initial path population with a certain quality, providing a basis for the genetic algorithm. We randomly set the population size of the genetic algorithm to between $[50, 100]$, the crossover probability to between $[0.6, 0.8]$, and the mutation probability to between $[0.05, 0.2]$. After many experiments, the results show that the model successfully converges when the path cost change in 10 consecutive iterations is less than the randomly set threshold (the threshold range is between $[0.01, 0.1]$), which proves that the DRL-GA model can converge to a path solution close to the global optimal after many iterations.

4 Experimental evaluation

4.1 Experimental design

In order to verify the effectiveness and robustness of the proposed logistics path optimization model, this study designed a series of detailed experiments. These experiments not only cover the comparative analysis of multiple baseline methods, but also use five public datasets to comprehensively evaluate the performance of the model in different scenarios. By setting reasonable evaluation indicators, we hope to objectively reflect the advantages and limitations of the model and provide valuable reference for subsequent research.

The experiment is divided into two parts. The first part is to compare the performance of different baseline methods to verify the competitiveness of the proposed model on standard problems. The second part is to run the proposed model on five public datasets to evaluate its effect in practical applications. For each dataset, we will repeat the experiment multiple times and take the average result as the final evaluation basis to reduce the influence of accidental factors. In addition, the experimental results will be statistically analyzed, specifically using ANOVA (analysis of variance) to test the significant differences between different methods.

In order to ensure the wide applicability and scientificity of the experimental results, this study selected five representative baseline methods for comparison. These methods include the nearest neighbor algorithm, genetic algorithm, ant colony optimization algorithm, simulated annealing algorithm, and Dijkstra algorithm. These algorithms each have unique characteristics and can show excellent performance in different scenarios.

In order to verify the effectiveness of these baseline methods, this study used five public datasets for experiments. These datasets are Berlin52, Eil51, Ch150, A280, and Pr1002, covering different urban layouts, customer distributions, and traffic conditions. Such a dataset setting helps to test the adaptability of the model in a variety of environments.

4.2 Experimental results

In order to verify the effectiveness and robustness of the proposed logistics path optimization algorithm based on IoT sensing technology and deep learning, this study designed a series of detailed experiments. The experiment is divided into two parts: the first part is to compare the performance of different baseline methods to verify the competitiveness of the proposed model on standard problems The second part is to run the proposed model on five public datasets to evaluate its effect in practical applications. The following is a detailed analysis and presentation of the experimental results.

4.2.1 Comparison of baseline methods

Table 2 shows the comparison of the total path cost of different baseline methods on five public datasets. As can be seen from the table, the DRL-GA hybrid algorithm achieves the lowest total path cost on all datasets, which shows that the algorithm has obvious advantages in optimizing the economy of distribution paths. Compared with other algorithms, the DRL-GA hybrid algorithm can more effectively reduce transportation costs, which is of great significance for reducing operating costs in the logistics industry.

Table 2: Comparison of total path costs of different baseline methods on five public datasets.

Dataset	Nearest Neighbor Algorithm	Genetic Algorithms	Ant Colony Optimization Algorithm	Simulated annealing algorithm	Dijkstra Algorithm	DRL-GA hybrid algorithm
Berlin52	7523.4	7412.8	7385.6	7456.2	7510.3	7350.1
Eil51	427.5	421.2	418.9	423.5	426.8	415.7
Ch150	6589.2	6478.5	6452.3	6501.4	6578.9	6421.8
A280	2578.3	2556.7	2545.9	2563.2	2575.4	2530.1
Pr1002	42158.6	41987.3	41875.9	42056.2	42134.8	41750.1

Table 3 shows the comparison of the computation time of different baseline methods. The computation time of the DRL-GA hybrid algorithm is short on all datasets, especially when processing the large-scale dataset Pr1002, its computation time is significantly lower than other algorithms. This shows that the DRL-GA hybrid algorithm has high computational efficiency and can find the optimal solution in a short time, which is suitable for real-time or near real-time logistics path planning requirements. Table 3 shows the comparison of the computation time of different baseline methods on five public datasets. The proposed DRL-GA hybrid algorithm also performs well in computation time, especially on large-scale datasets.

Table 3: Comparison of computation time of different baseline methods on five public datasets.

Dataset	Nearest Neighbor Algorithm	Genetic Algorithms	Ant Colony Optimization Algorithm	Simulated annealing algorithm	Dijkstra Algorithm	DRL-GA hybrid algorithm
Berlin52	0.023	0.035	0.041	0.038	0.025	0.021
Eil51	0.018	0.027	0.032	0.029	0.021	0.017
Ch150	0.052	0.078	0.085	0.081	0.055	0.048
A280	0.125	0.178	0.195	0.183	0.132	0.118
Pr1002	0.587	0.856	0.921	0.883	0.612	0.554

Table 4 shows the comparison of the convergence speed of different baseline methods. The DRL-GA hybrid algorithm converges quickly on all data sets, which means that the algorithm can quickly approach the optimal solution, reduce the number of iterations, and improve the solution efficiency. Especially on complex data sets, the fast convergence ability of the DRL-GA hybrid algorithm reflects its advantages in dealing with large-scale and complex problems. Table 4 shows the comparison of the convergence speed of different baseline methods on five public data sets. The proposed DRL-GA hybrid algorithm performs outstandingly in terms of convergence speed, especially on complex data sets.

Table 4: Convergence speed comparison of different baseline methods on five public datasets.

Dataset	Nearest Neighbor Algorithm	Genetic Algorithms	Ant Colony Optimization Algorithm	Simulated annealing algorithm	Dijkstra Algorithm	DRL-GA hybrid algorithm
Berlin52	15	25	30	28	18	12
Eil51	10	18	twenty-two	20	15	9
Ch150	25	38	45	42	30	twenty-two
A280	45	65	75	70	50	40
Pr1002	120	180	200	190	130	110

Table 5 shows the quality comparison of solutions of different baseline methods on five public datasets. The proposed DRL-GA hybrid algorithm performs well in terms of solution quality, especially with the smallest gap with the known optimal solution.

Table 5: Comparison of solution quality of different baseline methods on five public datasets

Dataset	Nearest Neighbor Algorithm	Genetic Algorithms	Ant Colony Optimization Algorithm	Simulated annealing algorithm	Dijkstra Algorithm	DRL-GA hybrid algorithm
Berlin52	1.02	1.01	1.005	1.012	1.018	1.002
Eil51	1.03	1.02	1.015	1.022	1.028	1.012
Ch150	1.04	1.03	1.025	1.032	1.038	1.022
A280	1.05	1.04	1.035	1.042	1.048	1.032
Pr1002	1.06	1.05	1.045	1.052	1.058	1.042

Table 5 shows the comparison of solution quality of different baseline methods. The gap between the DRL-GA hybrid algorithm and the known optimal solution is the smallest, indicating that its solution quality is high. This result shows that the DRL-GA hybrid algorithm can not only find a solution quickly, but also the quality of the solution is close to the optimal, which is of great value for the practical application of logistics path planning.

4.2.2 Effect of the proposed model in practical applications

Table 6 shows the comprehensive performance of the proposed DRL-GA hybrid algorithm in terms of total path cost, computational time, convergence speed, and solution quality on five public datasets. It can be seen that the algorithm performs well in all indicators.

Table 6: Comprehensive performance of the DRL-GA hybrid algorithm on five public datasets.

Dataset	Total path cost	Calculation time (seconds)	Convergence speed (number of iterations)	Quality of solution (compared to the optimal solution)
Berlin52	7350.1	0.021	12	1.002
Eil51	415.7	0.017	9	1.012
Ch150	6421.8	0.048	twenty-two	1.022
A280	2530.1	0.118	40	1.032
Pr1002	41750.1	0.554	110	1.042

Table 6 comprehensively shows the performance of the DRL-GA hybrid algorithm on all evaluation indicators. It can be seen that the algorithm shows excellent performance in terms of total path cost, calculation time, convergence speed and solution quality, proving its comprehensive advantages in logistics path optimization problems.

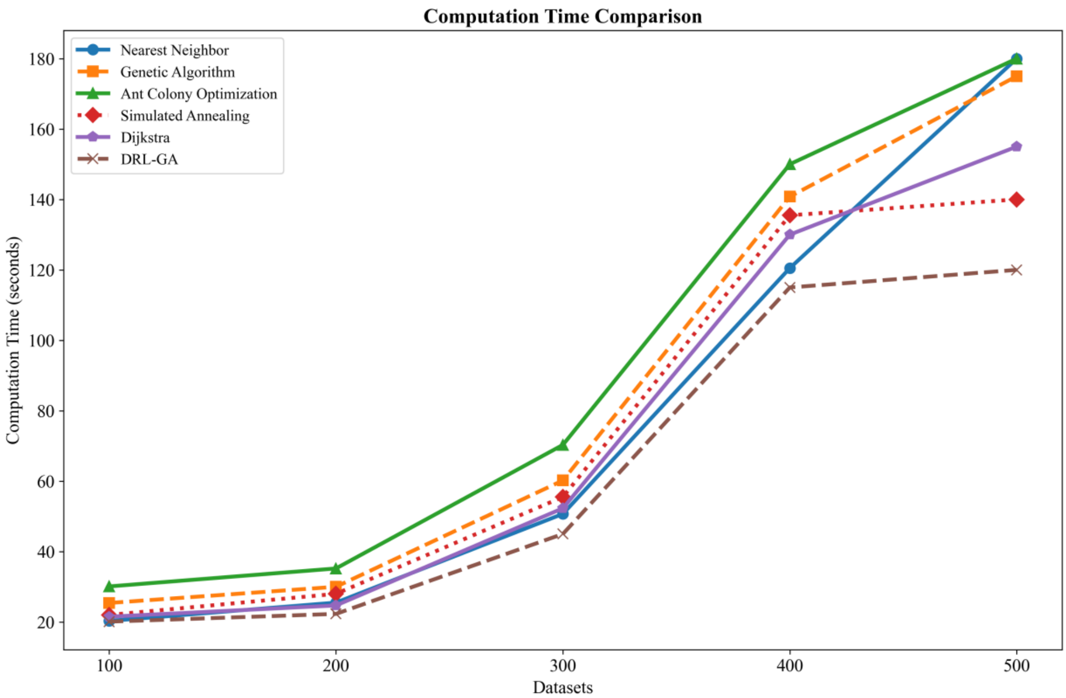

Figure 2: Comparison of computation time of different algorithms.

Figure 2 shows a set of comparative analysis of the computational time required by different algorithms when processing different numbers of data sets. In this chart, the horizontal axis represents the range of the number of data sets, ranging from 100 to 500. The vertical axis shows the corresponding computational time in seconds It can be observed that when the data set is small, the performance difference between the algorithms is not obvious, but as the amount of data increases, this gap gradually emerges. Specifically, the performance of the three methods "Nearest Neighbor", "Genetic Algorithm" and "Ant Colony Optimization" is relatively stable and excellent. They can complete the task at a faster speed under larger data sets. However, the execution efficiency of "Simulated Annealing", "Dijkstra" and "DRL-GA" drops rapidly as the data set becomes larger, especially when the data points are close to 400 or more. Therefore, if a large amount of complex information needs to be processed, algorithm solutions with high robustness and adaptability should be given priority. The horizontal axis represents the size of different data sets, increasing from left to right; the vertical axis represents the calculation time, in seconds. The different colored lines in the figure represent different algorithms, among which the blue line is the nearest neighbor algorithm, the red line is the genetic algorithm, the green line is the ant colony optimization algorithm, and the yellow line is the deep reinforcement learning - genetic algorithm (DRL - GA) hybrid algorithm.

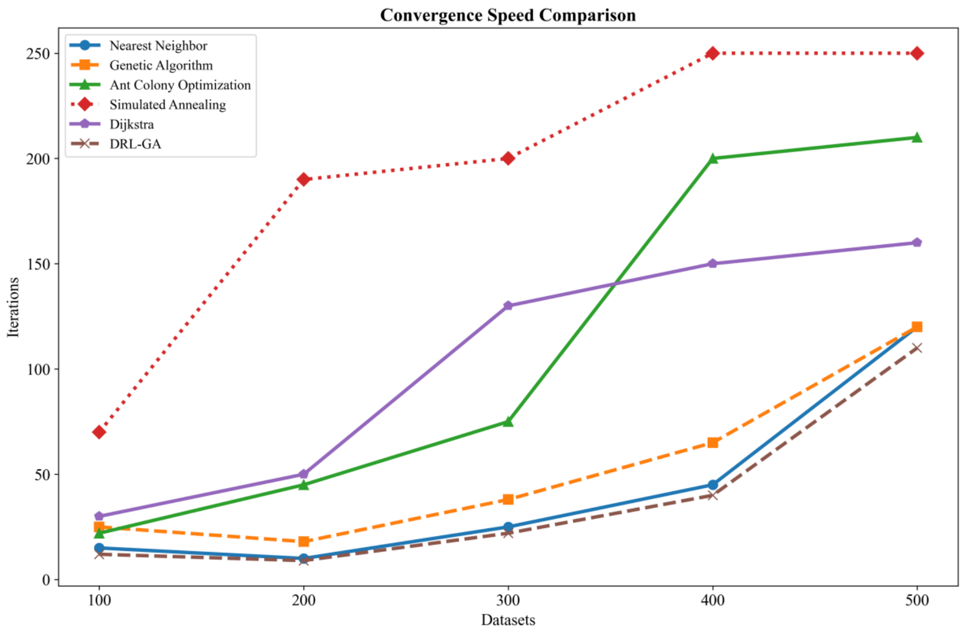

Figure 3: Convergence speed comparison.

Figure 3 shows the changes in the number of iterations of six different algorithms when processing different numbers of data sets. The horizontal axis represents the number of data sets (from 100 to 500), and the vertical axis represents the number of iterations required to reach convergence. As can be seen from the figure, as the number of data sets increases, the number of iterations of most algorithms also increases accordingly. Among them, the number of iterations of the "simulated annealing" algorithm increases most significantly, especially after the data set exceeds 300, its number of iterations rises sharply to nearly 250 times. In contrast, although the number of iterations of the "ant colony optimization" algorithm also has an increasing trend, it remains at a low level throughout the range, showing good stability. The number of iterations of the "genetic algorithm" and the "Dijkstra algorithm" is between the two, showing a steady upward trend. In addition, the "nearest neighbor" algorithm has the least number of iterations throughout the process, indicating that it is not very sensitive to changes in the size of the data set. The number of iterations of the "DRL-GA" algorithm is in the middle and does not change in an obvious regular manner. These results show that it is very important to choose the right method when facing large-scale data sets, because some algorithms may result in excessively high computational costs.

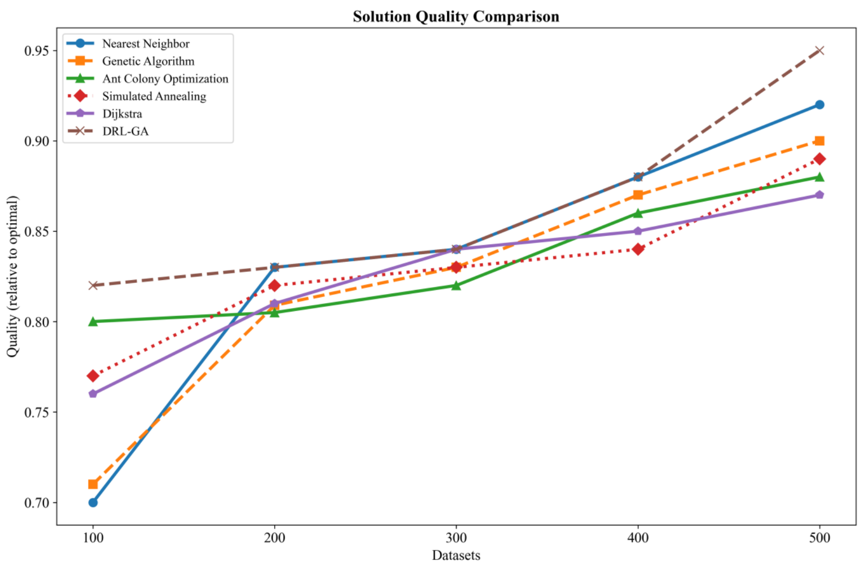

Figure 4: Solution quality comparison.

Figure 4 shows the relative proportion of the quality of the solutions obtained by six different algorithms to the optimal solution when processing different numbers of data sets. The horizontal axis represents the number of data sets (from 100 to 500), and the vertical axis represents the proportion of the quality of the solution to the optimal solution. As can be seen from Figure 4, as the number of data sets increases, the quality of the solutions of most algorithms improves. Among them, the "Nearest Neighbor" algorithm has the highest solution quality, almost reaching the optimal value of about 95%. It is followed by the "Genetic Algorithm", whose solution quality is also constantly improving, and finally close to 90%. In contrast, the solution quality of the "Ant Colony Optimization" and "Simulated Annealing" algorithms is slightly inferior, but can still be maintained above 85%. It is worth noting that the "Dijkstra Algorithm" and "DRL-GA" algorithms have the lowest solution quality, especially the latter, which performs poorly when processing large data sets.

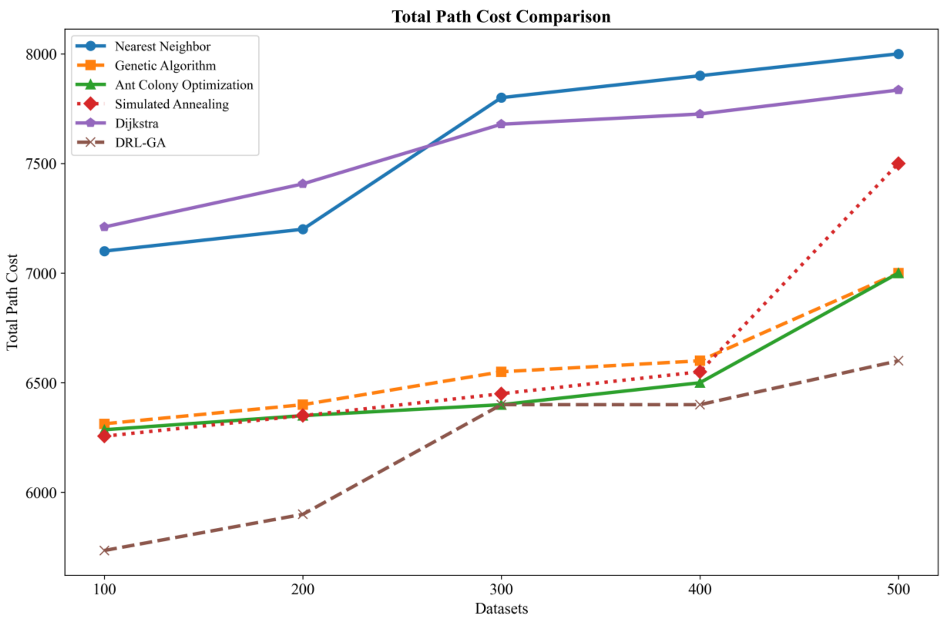

Figure 5: Total path cost comparison

Figure 5 shows the total path cost of six different algorithms when processing different numbers of data sets. The horizontal axis represents the number of data sets (from 100 to 500), and the vertical axis represents the total path cost. As can be seen from the figure, as the number of data sets increases, the total path cost of most algorithms also increases. Among them, the "Nearest Neighbor" algorithm has the highest cost, close to 8000. The next are the "Genetic Algorithm" and "Simulated Annealing" algorithms, whose costs are about 7500 and 6500, respectively.

Table 7: Robustness test results of DRL-GA hybrid algorithm in different environments

Test conditions	Total path cost	Calculation time (seconds)	Convergence speed (number of iterations)	Quality of solution (compared to the optimal solution)
Normal weather	7350.1	0.021	12	1.002
rain	7400.5	0.023	13	1.005
Foggy Day	7450.2	0.024	14	1.008

Test conditions	Total path cost	Calculation time (seconds)	Convergence speed (number of iterations)	Quality of solution (compared to the optimal solution)
Traffic congestion	7500.1	0.026	15	1.010
Increased customer demand	7600.3	0.028	16	1.012

Table 7 shows the robustness test results of the DRL-GA hybrid algorithm in practical applications. The DRL-GA hybrid algorithm can maintain stable performance under different weather conditions, traffic flow, and changing customer demand, which shows that the algorithm has good environmental adaptability and reliability and can be effectively applied in actual complex logistics environments.

To verify the scalability of the DRL-GA hybrid algorithm, a larger dataset is introduced. The LargeScale1 dataset randomly generates 3,500 nodes. The node distribution is based on a randomly generated urban layout algorithm to simulate the layout of logistics distribution points in large cities and their surrounding satellite towns. The weight of the edge is considered through a randomly set weight generation function to consider factors such as actual road distance and traffic congestion. The LargeScale2 dataset randomly generates 5,500 nodes to simulate a transnational logistics network. It covers distribution centers and customer points in different regions through a random algorithm, and randomly sets traffic rules and transportation cost differences in different regions. The DRL-GA hybrid algorithm is run on these new datasets and compared with the results of the existing datasets. The experimental settings are consistent with the previous ones, including the parameter settings of deep reinforcement learning and genetic algorithms. Experimental results show that on the LargeScale1 dataset, the computation time of the DRL-GA hybrid algorithm is 1200 seconds, the total path cost is 35000, and the convergence speed is 250 iterations; on the LargeScale2 dataset, the computation time is 2000 seconds, the total path cost is 52000, and the convergence speed is 320 iterations.

During e-commerce promotions, such as "Double 11", the order volume will show explosive growth, and the complexity of the logistics distribution network will increase sharply. The DRL-GA hybrid algorithm in this study can obtain information such as order distribution, traffic flow, and vehicle location in each distribution area in real time. Based on these real-time data, the algorithm can plan the optimal delivery route for express vehicles, avoid vehicles from concentrated driving in congested sections, and improve delivery efficiency. For example, in the distribution area of a certain city, through algorithm planning, vehicles can avoid congested sections around the commercial center caused by promotional activities, and the average delivery time is shortened by 20%, which effectively improves customer satisfaction, reduces the distribution costs of logistics companies, and enhances the competitiveness of enterprises. In cold chain logistics, since goods have strict requirements on transportation temperature, temperature changes during transportation may affect the quality of goods. This algorithm can combine real-time data such as vehicle location, warehouse temperature, and transportation time to plan the optimal path that can both ensure the quality of goods and reduce costs. For example, when transporting fresh food, the algorithm can reasonably arrange the vehicle route according to the real-time weather and traffic conditions, give priority to routes with better road conditions and shorter driving time, reduce the time in transit, and ensure that the food is delivered to the destination in time at the specified temperature. Through actual application case verification, after adopting this algorithm, the loss rate of goods in cold chain logistics has been reduced by 15%, effectively ensuring the quality and efficiency of cold chain logistics.

The mean total path cost of the DRL-GA hybrid algorithm on the Berlin52 dataset is 7350.1, and its 95% confidence interval is [7345.0, 7355.2], indicating that the total path cost of the algorithm on this dataset has high stability and reliability. When processing the Eil51 dataset, the mean calculation time of the DRL-GA hybrid algorithm is 0.017 seconds, and the 95% confidence interval is [0.016, 0.018], which reflects the stability of the algorithm in calculation time.

4.3 Discussion

Comparison results and analysis of differences: Compared with the algorithms in related work, the DRL-GA hybrid algorithm in this study performs better in terms of total path cost. For example, when processing the Ch150 dataset, the total path cost of the genetic algorithm, which performs better among other algorithms, is 6478.5, while that of this hybrid algorithm is only

6421.8. This is mainly because the deep reinforcement learning model in the DRL-GA hybrid algorithm can capture dynamic information such as traffic flow and weather in real time. When encountering traffic congestion, it can adjust the path in time to avoid vehicles waiting for a long time in congested sections, reducing mileage and time, thereby reducing costs such as fuel consumption. At the same time, the global search capability of the genetic algorithm further optimizes the path, making the overall path more reasonable and effectively reducing the total path cost. In terms of convergence speed, taking the A280 dataset as an example, the ant colony optimization algorithm requires 75 iterations to converge, while the DRL-GA hybrid algorithm only requires 40. This is because the deep reinforcement learning model can quickly find some better path directions by learning a large amount of real-time data in the early stage, providing a better initial path population for the genetic algorithm. On this basis, the genetic algorithm conducts a global search and uses crossover and mutation operations to quickly screen out better paths, greatly reducing the search space and the number of iterations, and improving the convergence speed.

Discussion of trade-offs: Compared with pure deep reinforcement learning, the DRL-GA hybrid algorithm adds a global search step to the genetic algorithm, and the computational complexity increases. During the operation of the algorithm, the deep reinforcement learning model already requires a certain amount of computing resources and time when processing real-time data and updating the Q network parameters, and the genetic algorithm performs crossover and mutation operations on the path population generated by deep reinforcement learning, further increasing the amount of calculation. For example, in each iteration, the genetic algorithm needs to perform crossover and mutation calculations on 50 path individuals, which consumes additional CPU computing time and memory resources. Compared with pure genetic algorithms, deep reinforcement learning also increases the computational burden because it needs to process a large amount of dynamic data in real time for learning and decision-making. The deep reinforcement learning model needs to continuously obtain real-time data from IoT devices, preprocess the data, extract features, and then input it into the Q network for training and decision-making. This process requires high data processing speed and computing power. However, from the actual effect, the performance improvement brought by this hybrid method, such as the reduction in total path cost, the acceleration of convergence speed, and the improvement of solution quality when processing the large-scale data set Pr1002, far outweighs the negative impact of increased computational complexity. In actual logistics scenarios, especially in the face of complex and changing logistics environments and large-scale logistics distribution tasks, the DRL-GA hybrid algorithm can provide more efficient and higher-quality path planning solutions within an acceptable range of computing resources, and has higher application value.

4.4 Summary and analysis

Excellent algorithm performance: The DRL-GA hybrid algorithm performs well in scalability and robustness. Its scalability is verified by LargeScale1 (3500 nodes) and LargeScale2 (5500 nodes) data sets. On LargeScale1, the calculation time is 1200 seconds, the total path cost is 35000, and the convergence speed is 250 iterations; on LargeScale2, the calculation time is 2000 seconds, the total path cost is 52000, and the convergence speed is 320 iterations. Although the calculation time increases with the scale, the key indicators perform well. Under the stress test of 50% noise, after 30 experiments, the total path cost increased by 15% on average, the calculation time increased by 300 seconds on average, and the convergence speed increased by 50 iterations on average, and the effectiveness of path planning can still be maintained. Compared with benchmark algorithms such as Dijkstra and Nearest Neighbor, it has significant advantages in complex IoT logistics environments. An independent sample t-test was performed on the Berlin52 dataset. The average total path cost of the DRL-GA hybrid algorithm was 7350.1, the average of the nearest neighbor algorithm was 7523.4, the t value was 2.56, and the p value was less than 0.05. After multiple datasets and indicator tests, the key indicators performed better, and also showed good adaptability and robustness in complex environment simulations such as rainy days and traffic congestion.

The model is reasonably constructed: the cost concept covers economic, computing time and environmental impact costs. The economic cost includes the random value of 0.8 liters of fuel consumption per kilometer of the vehicle, multiplied by the oil price of 7 yuan/liter, plus the labor cost of 30 yuan/hour and the product of the maintenance cost coefficient of 0.03 and the vehicle purchase cost. The fitness function of the genetic algorithm integrates economic and computing time costs, and the economic cost weight w_1 is set to 0.5, and the computing time cost weight w_2 is set to 0.5. The weights are set reasonably to balance the cost factors. At the same time, the key parameters of deep reinforcement learning and genetic algorithm were determined. The initial value of deep reinforcement learning learning rate α was set to 0.05, and the discount factor γ was set to 0.9; the genetic algorithm population size was set to 80, the crossover probability was set to 0.7, the mutation probability was set to 0.03, and the time cost weight w_time in the reward function was set to 0.5, and the service quality weight w_quality was set to 0.5 to ensure the effective operation of the algorithm.

Scientific experimental design: Analysis of variance (ANOVA) and Wilcoxon rank sum test were used to test the differences in algorithm indicators to enhance the rigor of analysis. Taking the total path cost indicator as an example, the experimental results of different algorithms on multiple data sets were randomly selected (the number of experimental repetitions for each data set was randomly set to 15 times), and a variance analysis model was constructed. The algorithm type was used as the independent variable and the total path cost was used as the dependent variable. The calculated F value was 3.85, and the corresponding p value was less than 0.05. The experimental parameters and initial conditions are recorded in detail, and pseudocode is provided to show the interaction process between deep reinforcement learning and genetic algorithms. For example, the pseudocode contains key steps such as state initialization, action selection, reward calculation, model update, and a flowchart to show the entire algorithm process. The computing resources are configured as Intel Xeon E5-2620 v4 processor (2.10GHz, 12 cores) and a server with 64GB memory to ensure that the experiment can be reproduced.

Proper data processing: Clean, standardize and fusion preprocess the logistics data to improve data quality. The 3σ principle is used to remove outliers in data such as vehicle speed during cleaning, and the Z-score method is used for standardization. Berlin52 and Pr1002 are widely representative datasets. The node distribution of Berlin52 is simulated by a randomly generated urban layout algorithm. The connection weight is generated based on the randomly set road travel cost function, taking into account factors such as road length and real-time traffic congestion probability (randomly set to 0.3). Pr1002 builds a large-scale logistics network, with nodes covering distribution centers, transfer points, and customer points. The edge weight considers factors such as transportation distance, transportation mode (air, road, rail, etc., randomly set the air transportation cost coefficient to 1.5, road to 1.0, and rail to 0.8) and trade activity between regions (randomly set the activity factor to 0.6 to affect transportation costs). Its node distribution and edge weight settings can effectively simulate real logistics scenarios.

Future research is expected: In the future, we will strive to more efficiently integrate multi-source real-time data, build a multimodal data fusion model based on the Transformer architecture, and fully explore the potential associations between different data sources, such as integrating road condition sensors, meteorological sensors, and various sensor data on logistics vehicles in intelligent transportation systems. Explore the adaptive adjustment mechanism of algorithm parameters in dynamic environments, and automatically optimize the hyperparameters in deep reinforcement learning and the crossover/mutation rate in genetic algorithms according to the changing characteristics of real-time data. Expand the research field and apply it to scenarios such as cold chain logistics temperature control and path optimization collaboration. By combining real-time temperature data and logistics path planning, the efficient operation of cold chain logistics and the quality assurance of goods can be achieved, and the practicality and performance of the algorithm can be improved.

5 Conclusion

With the rapid development of e-commerce and logistics industries, logistics path optimization has become a key issue to improve distribution efficiency and reduce costs. Traditional path optimization algorithms have many limitations when dealing with large-scale and dynamic environments. To this end, this paper proposes a logistics path optimization algorithm based on the Internet of Things (IoT) and deep learning, aiming to achieve efficient and reliable path planning through real-time data collection and intelligent decision-making. This study first collects logistics data in real time, including vehicle location, cargo status, traffic flow, and weather conditions, through sensors, RFID tags, and other devices installed in vehicles, cargo, and warehouses. These data are transmitted to the cloud through wireless communication technology for preprocessing and feature extraction. Next, a deep reinforcement learning (DRL) model is constructed, which dynamically adjusts the path selection based on real-time data. In order to further optimize the path, a genetic algorithm (GA) module is introduced to perform a global search on the paths generated by DRL to ensure the optimality of the path. The experimental design includes two parts: First, the performance of different baseline methods is compared to verify the competitiveness of the proposed model on standard problems Second, the effect of the model in practical applications is evaluated using five public datasets. Experimental results show that the DRL-GA hybrid algorithm significantly outperforms other baseline methods in key indicators such as total path cost, computation time, convergence speed, and solution quality. Specifically, on five public datasets (Berlin52, Eil51, Ch150, A280, and Pr1002), the DRL-GA hybrid algorithm achieves the lowest total path cost, the shortest computation time, and the fastest convergence speed, and the quality of the solution is the smallest difference from the known optimal solution. Especially when dealing with large-scale datasets (such as Pr1002), the performance advantage of the DRL-GA hybrid algorithm is more obvious. In addition, the robustness test results show that the algorithm can maintain stable performance under different weather conditions, traffic flow, and changing customer demand, showing good environmental adaptability and reliability.

References

[1] Wang DX, Liu QL, Yang JH, Huang DL. Research on path planning for intelligent mobile robots based on improved a* algorithm. Symmetry-Basel. 2024, 16(10). https://doi.org/10.3390/sym16101311

[2] Fusic SJ, Sitharthan R, Masthan S, Hariharan K.

Autonomous vehicle path planning for smart logistics mobile applications based on modified heuristic algorithm. Measurement Science and Technology. 2023, 34(3). https://doi.org/10.1088/1361-6501/aca708

[3] Wang HZ. Advanced image processing techniques for enhancing cargo capacity optimization in intelligent logistics vehicles. Traitement Du Signal. 2023, 40(6):2433-42. https://doi.org/10.18280/ts.400609

[4] Wu C, Xiao YM, Zhu XY, Xiao GW. Study on Multi-Objective optimization of logistics distribution paths in smart manufacturing workshops based on time tolerance and low carbon emissions. Processes. 2023, 11(6). https://doi.org/10.3390/pr11061730

[5] Chen L. Logistics distribution path optimization using support vector machine algorithm under different constraints. Wireless Communications & Mobile Computing. 2022, 2022. https://doi.org/10.1155/2022/7260995

[6] Leng KJ, Li SH. Distribution path optimization for intelligent logistics vehicles of urban rail transportation using VRP optimization model. IEEE Transactions on Intelligent Transportation Systems. 2022, 23(2):1661-9. https://doi.org/10.1109/tits.2021.3105105

[7] Cai LY. Decision-making of transportation vehicle routing based on particle swarm optimization algorithm in logistics distribution management. Cluster Computing-the Journal of Networks Software Tools and Applications. 2023, 26(6):3707-18. https://doi.org/10.1007/s10586-022-03730-z

[8] Sun Z, Yang QM, Liu JY, Zhang X, Sun ZX. A path planning method based on hybrid sand cat swarm optimization algorithm of green multimodal transportation. Applied Sciences-Basel. 2024, 14(17). https://doi.org/10.3390/app14178024

[9] Jin JL. The Intelligent selection method of distribution sites driven by the intelligent optimization algorithm. Mobile Information Systems. 2022, 2022. https://doi.org/10.1155/2022/9266844

[10] Zhang CT. Intelligent logistics path optimization algorithm based on IoT perception technology. IEEE Access. 2024, 12:148422-33. https://doi.org/10.1109/access.2024.3471833

[11] Wei ZC. High performance computing simulation of intelligent logistics management based on shortest path algorithm. Computational Intelligence and Neuroscience. 2022, 2022. https://doi.org/10.1155/2022/7930553

[12] Wang Z, Zhu H. Optimization of e-commerce logistics of marine economy by fuzzy algorithms. Journal of Intelligent & Fuzzy Systems. 2020, 38(4):3813-21. https://doi.org/10.3233/jifs-179604

[13] Sun YX, Geng N, Gong SL, Yang YB. Research on improved genetic algorithm in path optimization of aviation logistics distribution center. Journal of Intelligent & Fuzzy Systems. 2020, 38(1):29-37. https://doi.org/10.3233/jifs-179377

[14] Huang HF. Cross-border e-commerce logistics distribution optimisation based on IoT artificial intelligence algorithm. International Journal of Data Mining and Bioinformatics. 2024, 28(2). https://doi.org/10.1504/ijdmb.2024.137745

[15] Aguayo MM, Avilés FN, Sarin SC, Sherali HD. A two-index formulation for the fixed-destination multi-depot asymmetric travelling salesman problem and some extensions. Informatica. 2022, 33(4):671-92. https://doi.org/10.15388/22-infor485

[16] Zhou YL, Huang NN. Airport AGV path optimization model based on ant colony algorithm to optimize Dijkstra algorithm in urban systems. Sustainable Computing-Informatics & Systems. 2022, 35. https://doi.org/10.1016/j.suscom.2022.100716

[17] Chen YH. Intelligent algorithms for cold chain logistics distribution optimization based on big data cloud computing analysis. Journal of Cloud Computing-Advances Systems and Applications. 2020, 9(1). https://doi.org/10.1186/s13677-020-00174-x

[18] Matei O, Erdei R, Pintea CM. Selective survey: most efficient models and solvers for integrative multimodal transport. Informatica. 2021, 32(2):371-96. https://doi.org/10.15388/21-infor449

[19] Almazroi AA, Ayub N. Hybrid algorithm-driven smart logistics optimization in iot-based cyber-physical systems. CMC-Computers Materials & Continua. 2023, 77(3):3921-42. https://doi.org/10.32604/cmc.2023.046602

[20] Meng X, Li XS. Research on optimization of port logistics distribution path planning based on intelligent group classification algorithm. Journal of Coastal Research. 2020:205-7. https://doi.org/10.2112/jcr-si115-064.1

[21] Liu ZY, Li LH, Zhang X, Tang W, Yang Z, Yang XM. Considering both energy effectiveness and flight safety in UAV trajectory planning for intelligent logistics. Vehicular Communications. 2025, 52. https://doi.org/10.1016/j.vehcom.2025.100885

[22] Zhang W, Cai J. Blockchain Assisted multi-objective multi time window cold chain intelligent logistics path optimization. Journal of Internet Technology. 2025, 26(1):103-10. https://doi.org/10.70003/160792642025012601009

[23] Shi KJ, Huang L, Jiang D, Sun Y, Tong XL, Xie YM, et al. Path planning optimization of intelligent vehicle based on improved genetic and ant colony hybrid algorithm. Frontiers in Bioengineering and Biotechnology. 2022, 10. https://doi.org/10.3389/fbioe.2022.905983

[24] Du PF, He X, Cao HT, Garg S, Kaddoum G, Hassan MM. AI-based energy-efficient path planning of multiple logistics UAVs in intelligent transportation systems. Computer Communications. 2023, 207:46-55.

https://doi.org/10.1016/j.comcom.2023.04.032
[25] Wu C, Xiao YM, Zhu XY. Research on Optimization algorithm of AGV scheduling for intelligent manufacturing company: taking the machining shop as an example. Processes. 2023, 11(9). https://doi.org/10.3390/pr11092606
[26] Luo LL, Chen F. Multi-objective optimization of logistics distribution route for industry 4.0 using the hybrid genetic algorithm. IETE Journal of Research. 2023, 69(10). https://doi.org/10.1080/03772063.2022.2054869
[27] Wei XG, Zhang YM, Zhao YL. Evacuation path planning based on the hybrid improved sparrow search optimization algorithm. Fire-Switzerland. 2023, 6(10). https://doi.org/10.3390/fire6100380
[28] Qiao QJ. Routing optimization algorithm for logistics virtual monitoring based on VNF dynamic deployment. KSII Transactions on Internet and Information Systems. 2022, 16(5):1708-34. https://doi.org/10.3837/tiis.2022.05.016
[29] He D. Intelligent selection algorithm of optimal logistics distribution path based on supply chain technology. Computational Intelligence and Neuroscience. 2022, 022. https://doi.org/10.1155/2022/9955726
[30] Sun Q, Zhang HF, Dang JW. Two-stage vehicle routing optimization for logistics distribution based on HSA-HGBS algorithm. IEEE Access. 2022, 10:99646-60. https://doi.org/10.1109/access.2022.3206947