

Retraction

Retracted: Intelligent Recognition Method of Vehicle Path with Time Window Based on Genetic Algorithm

Security and Communication Networks

Received 7 April 2022; Accepted 7 April 2022; Published 17 May 2022

Copyright © 2022 Security and Communication Networks. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Security and Communication Networks has retracted the article titled “Intelligent Recognition Method of Vehicle Path with Time Window Based on Genetic Algorithm” [1], due to concerns with the authenticity of the data. It was found that previous versions of this submission contained a figure unrelated to the topic of the paper and this graph was subsequently identified within several other submissions, all with accompanying text claiming to have generated the graph. A number of these submissions were rejected from the journal; however, 6 were published and have now been retracted from *Security and Communication Networks* and *Scientific Programming* [2–6].

The authors responded to explain that an author from one of the identified submissions had provided copy editing for their manuscript and introduced the graph and accompanying text in error.

The authors were unable to provide copies of correspondence to support their claim or the raw data from their study. The authors’ explanation did not satisfy the concerns of the editorial board, and the article is therefore being retracted due to concerns with the reliability of the data.

The authors do not agree to the retraction.

References

- [1] L. Huo, “Intelligent Recognition Method of Vehicle Path with Time Window Based on Genetic Algorithm,” *Security and Communication Networks*, vol. 2021, Article ID 3614291, 11 pages, 2021.
- [2] L. Keshun, Z. Xijun, and Z. Xing, “Research on Analysis and Classification of Vulnerability of Electromagnetic Pulse with a STM32 Single-Chip Microcomputer,” *Scientific Programming*, vol. 2021, Article ID 6836158, 10 pages, 2021.
- [3] L. Zhuang, Y. Qi, and F. Zhang, “Braking Control System of Oilfield Minor Repair Machine Based on Wireless Sensor

Network,” *Security and Communication Networks*, vol. 2021, Article ID 6966041, 14 pages, 2021.

- [4] L. Fan, M. Xia, P. Huang, and J. Hu, “Research on Educational Information Platform Based on Cloud Computing,” *Security and Communication Networks*, vol. 2021, Article ID 3109473, 11 pages, 2021.
- [5] L. Yan, “Research on Crude Oil Trade Procurement Model Based on DEA-Malmquist Algorithm,” *Scientific Programming*, vol. 2021, Article ID 6360439, 10 pages, 2021.
- [6] G. Yin, R. Qi, J. Zhao, X. Zhao, and J. Xu, “Optimization Method of Sports Service Network Node Layout Based on Network Communication,” *Security and Communication Networks*, vol. 2021, Article ID 3013135, 8 pages, 2021.

Research Article

Intelligent Recognition Method of Vehicle Path with Time Window Based on Genetic Algorithm

Lina Huo 

School of Mathematics and Statistics, Yulin University, Yulin 719000, China

Correspondence should be addressed to Lina Huo; huolina@yulinu.edu.cn

Received 30 May 2021; Revised 26 June 2021; Accepted 12 August 2021; Published 20 August 2021

Academic Editor: Chi-Hua Chen

Copyright © 2021 Lina Huo. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Based on particle filter and improved cuckoo genetic algorithm, an algorithm for intelligent vehicle path recognition with a time window is designed. Particle filter (PF) is an influential visual tracking tool; it relies on the Monte Carlo Chain framework and Bayesian probability, which are essential for intelligent monitoring systems. The algorithm first uses particle filters for visual tracking and then obtains the current operating environment of the vehicle, then performs cluster analysis on customer locations, and finally performs path recognition in each area. The algorithm not only introduces particle filters, which are advanced visual tracking, but also improves the cuckoo search algorithm; when the bird's egg is found by the bird's nest owner, it needs to randomly change the position of the entire bird's nest, which speeds up the search speed of the optimal delivery route. Analyze and compare the hybrid intelligent algorithm and the cuckoo search algorithm. Finally, the international standard test set Benchmark Problems is used for testing. The experimental outcomes indicated that the new hybrid intelligent approach is an effective algorithm for handling vehicle routing tasks with time windows.

1. Introduction

After 60 years of research and development, the vehicle path planning problem has changed from the original simple vehicle scheduling problem to a complex system problem [1]. The initial vehicle path planning problem can be described as follows: there are a starting point and several customer points, the geographical location and needs of each point are known, and how to plan the optimal path under various constraints, so that it can serve each customer point and return to the starting point at the end [2]. By imposing different constraints and changing the optimization goal, different types of vehicle path planning problems can be derived. Meanwhile, the vehicle path planning problem is a typical NP-hard problem, and its precise methods deal with the issue on a small scale, so the heuristic algorithm has attracted more attention recently [3].

RPTW (vehicle routing problem with time window) set a strong window constraint to all required nodes. Considering the NP-hard nature of the VRPSTW problem, accurate

algorithms are no longer suitable for solving problems of a certain scale, and the applicability of heuristic algorithms for solving large-scale VRP problems is prominent and has been widely recognized by the academic community, and many scholars improved the basis of traditional algorithms [4]. Fan et al. [5] applied variable-field descent search to the disturbance of the particle swarm algorithm to qualify the search outcomes of the proposal and applied it to solve the problem of simultaneous vehicle routing. Guo et al. [6] aimed at the vehicle routing problem under the background of emergency logistics and introduced the crowding factor in the artificial fish swarm algorithm into the ant one to guide aggregation of the ant colony. It is worth noting that the latter one is more suitable to handle VRP and its variants for the sake of computational complexity and convergence effectiveness, and the quality of the solution does not depend on the initialization of the solution and has good robustness [7]. However, due to the strong self-organization and forward feedback of the former one, it shows good convergence and is easy to fall into the local optimum. Therefore, some scholars have introduced domain search and elite retention

strategy or mixed other classics [8, 9]. Heuristic algorithm has obvious advantages over improved ant colony algorithm [10, 11]. It is found that the topology of a data communication network roles as an important factor in its security robustness against attack. In such networks, by changing the topology, the security robustness against the intentional attack that aims at bringing down network nodes may vary. This bringing down is a kind of destruction and interruption threat that attacks the availability of the network (i.e., attack on network resources and links). Recently, for the starting up algorithms for emerging groups, many scholars have considered applying them to their respective research fields [12]. Among them, the firefly algorithm is used in many continuous optimization problem, showing better performance [13]. It is found that the topology of a data communication network roles as an important factor in its security robustness against attack. In such networks, by changing the topology, the security robustness against the intentional attack that aims at bringing down network nodes may vary. This bringing down is a kind of destruction and interruption threat that attacks the availability of the network (i.e., attack on network resources and links). Due to the discrete nature of VRP, encoding and decoding of firefly are a key link. There has been literature that have considered this problem and improved the algorithm. Hua and Juncheng [14] made a discrete improvement to the firefly algorithm for solving continuous problems to adapt to the solution of open vehicle path problems. Existing studies have shown that the firefly algorithm can search globally well, but it has a strong constraint, and the algorithm has poor robustness and convergence. Combining the advantages and disadvantages of the above-mentioned ant colony algorithm and firefly algorithm, it provides ideas for the design of a hybrid algorithm. Based on this, considering the complementary advantages of the two algorithms, combine the two to design a hybrid algorithm so as to keep the effectiveness of the methods and improve the accuracy of the solution.

The contribution of our method can be described as follows:

- (1) Based on particle filter and improved cuckoo genetic algorithm, an algorithm for intelligent vehicle path recognition with time window is designed

- (2) The energy balanced clustering routing algorithm based on energy and location is used to optimize and balance the energy consumption distribution of nodes
- (3) In the straight-line and curve training path, this method has significant accuracy in recognizing intelligently the vehicle paths

This paper is divided into 5 sections. Section 2 gives a brief introduction to VRPSTW. Our proposal can be described in Section 3 in detail. Section 4 describes the experimental results. Section 5 concludes this paper with contributions, limitations, and future works.

2. VRPSTW

2.1. Background. Based on the terminal distribution in the logistics system, it can be attributed to the vehicle path problem with soft time windows (VRPSTW) [15]. Specifically, a material dispatch center is equipped with a sufficient number of homogeneous vehicles, and there is a set of customer demand points in the road network. Each customer's demand and service duration must be met. Consider meeting the customer service time window as much as possible. Reasonably plan the vehicle path to optimize the cost. The problems and basic assumptions that need to be solved [16] are as follows: (1) the material dispatch center starts and returns to the dispatch center after serving a number of customer demand points; (2) each customer point is merely estimated by one, once visited [17], and the demand cannot exceed the vehicle capacity; (3) the vehicle is the same model and the total transportation cannot exceed the center capacity limit; and (4) the coordinates, demand, and service duration of each customer are known and there is a service time window limit [18].

2.2. Mathematical Model. Transmission network $G = (V, A)$, the main parameters, and related variables are in Table 1.

Refer to the optimization direction of the traditional vehicle tracing tasks, the intention is to decrease the overall cost of delivery, and build a vehicle model with soft time windows:

$$\min z = \sum_{i \in V} \sum_{j \in V} \sum_{k \in K} x_{ijk} d_{ij} c_k + \sum_{j \in V_0} \sum_{k \in K} x_{0jk} f_k + \sum_{i \in V_0} CE_i \max\{(e_i - T_i), 0\} + \sum_{i \in V_0} CL_i \max\{(T_i - l_i), 0\}, \quad (1)$$

$$\text{s.t.} \quad \sum_{i \in V} \sum_{k \in K} x_{ijk} = 1, \quad \forall j \in V_0, \quad (2)$$

$$\sum_{j \in V} \sum_{k \in K} x_{ijk} = 1, \quad \forall i \in V_0,$$

$$\sum_{i \in V} \sum_{j \in V} x_{ijk} \leq |S| - 1, \quad \forall k \in K, |S| \geq 2, \quad (3)$$

$$\sum_{j \in V_0} x_{0jk} = \sum_{i \in V_0} x_{xi0k} = 1, \quad \forall k \in K, \quad (4)$$

TABLE 1: Model parameters and variable.

	Symbol	Meaning
Set	V	Note set, $V = \{1, 2, \dots, n\}$, 1 as dispatch center
	V_0	$V_0 = V/\{1\}$ gathering for customer needs
	A	Collection of paths, $A = \{(i, j): i, j \in V, i \neq j\}$
	S	Any subset of customer demand points, $S \subseteq V_0$
	K	Vehicle assembly, $K = \{k 1, 2, \dots, m\}$, k for delivery vehicles
	K	Vehicle assembly, $K = \{k 1, 2, \dots, m\}$, k for delivery vehicles
Parameters	Q	Vehicle capacity
	q_i	Material demand at the demand point
	u_i	Demand point $i \in V_0$ duration of service
	d_{ij}	$i \neq j \in V_0$ route $(i, j) \in A$ mileage
	f_k	Car $k \in K$ fixed start-up costs
	t_{ijk}	The travel time of vehicle k between $ii \in V$ path $(i, j) \in A$
	C	Control parameter, a very large integer
	c_k	The unit travel distance cost of the vehicle k
	CE_i	Arrive earlier than the time window, penalty cost per unit time
	CL_i	Arrival later than the time window, penalty cost per unit time
Variable	e_i	The lower limit of the time window of demand point i
	l_i	The upper limit of the time window of demand point i
	T_i	The actual arrival of the vehicle at the demand point $i \in V_0$
	x_{ijk}	Decision variable, when node $i \neq j \in V_0$ path between $(i, j) \in A$ by vehicle $k \in K$. The value is 1; otherwise 0.

$$\sum_{i \in V} x_{ihk} - \sum_{j \in V} x_{hjk} = 0, \quad \forall k \in K, h \in V_0, \quad (5)$$

$$\sum_{i \in V_0} \sum_{j \in V} x_{ijk} q_i \leq Q_k, \quad \forall k \in K, \quad (6)$$

$$T_{ik} + u_i + t_{ijk} \leq T_{jk} + C(1 - x_{ijk}), \quad \forall i, j \in V_0, k \in K, x_{ijk} \in \{0, 1\}, \forall i, j \in V, k \in K, Q_k, q_i, u_i > i \in V_0, k \in K. \quad (7)$$

Equation (1) is the objective function for minimizing the distribution cost [19]; equations (2) and (3) are that each customer demand point is only served by one car and visited only once; equation (4) is the subloop elimination constraint; equation (5) is the automobiles away from the dispatching center and finally going back to the dispatching zones; equation (6) is the node balance constraint; equation (7) is the vehicle capacity limit; and equation (7) is the vehicle from serving the previous demand point to visiting the next demand point. Time conditions to be satisfied.

In the last three years, many methods are proposed to handle the problem of intelligent recognition method of vehicle path with time window based on different network structures; here, we introduced three outstanding methods such as GNB (graph neural network based method) [20], RNB (Reset neural network based method) [21], and CNB (Conventional neural network based method) [22], which can be used to solve the related works taking different kinds of network structures. Adaptive Moment Estimation (Adam) is essentially RMSprop with momentum term. It uses the first-order Moment Estimation and second-order Moment Estimation of the gradient to dynamically adjust the learning rate of each parameter. The advantage of Adam is that after bias correction, the learning rate of each iteration has a definite range, making the parameters relatively stable.

NADAM is similar to ADAM with Nesterov momentum term. GNB is a graph-based network that builds connections between different risk nodes. And RNB uses a specific loss structure to keep the similarity of real and predicted crafts design. CNB is the basic model that needs more computation consuming to obtain the desiring performances. However, these methods have their disadvantages, respectively. GNB is too slow, RNB is so complicated, and CNB also needs more spaces. Compared with these methods, our proposal can work well effectively for path identification by using the evolving algorithm which does not utilize the structure of the network and is not time-consuming at all.

In this paper, we utilize the entropy loss function to build the model for our research problems. It can be defined as $\text{loss}(x, y) = \sum_{i=1}^n -p_i \log(1 - p_i)$, where x and y are represented as the real arts and crafts' score and difficulty and y means the predicted score and difficulty of our proposal. P_i means the probability of them when they are similar. The bigger the value of the loss, the worse our proposal performed.

And compared with the three methods, our proposal can deal with the problems easily and we also need a smaller computation space to build our model. However, our model may obtain a relatively lower accuracy than others sometimes which may lead to prediction instability.

3. Hybrid Ant Colony Algorithm for Solving VRPSTW

The RPSTW problem is a component of the classic VRP problem [23] and is regarded as the NP-hard problem. Due to the large scale of solving this problem, accurate algorithms cannot solve the exponential explosion problem [24], and the desired outcome is hard to access. The traditional heuristic algorithm for the VRP problem, such as genetic algorithm, ant colony algorithm, and many other methods, show good results [25–28]. Thus, a new heuristic algorithm is designed to handle the VRPSTW problem.

3.1. Ant Colony Algorithm. Ant colony algorithm (ACO) is a bionic algorithm that imitates the process of finding food by real ants [29–32]. Its basic principle is that ants calculate each pheromone concentration and heuristic information on the path (i, j) at time t . Transition probability $P_{ij}^s(t)$ of alternative transition node is

$$P_{ij}^s(t) = \begin{cases} \frac{[T_{i,j}(t)]^\alpha \cdot \eta_{i,j}^\beta}{\sum_{s \in \text{allowed}_s} [T_{i,j}(t)]^\alpha \cdot \eta_{i,s}^\beta}, & j \in \text{allowed}_s, \\ 0, & \text{otherwise,} \end{cases} \quad (8)$$

where α and β are two parameters which adjust the pheromone importance and heuristic information importance, respectively. $\text{Allowed}_s = V/\{\text{tabus}\}$ is the set of nodes that ants are allowed to access in the next step, and tabus is the set of nodes that ants have visited, that is, the path Taboo table.

The greater the heuristic function $\eta_{ij}(t)$, the greater the probability of the ant choosing node j . When we enlarged the value of iterations the pheromone concentration τ_{ij} of the path (i, j) will continue to be superimposed, and the remaining pheromone will continue to volatilize, then the pheromone $\tau_{ij}(t+1)$ concentration of the path (i, j) at time $(t+1)$ will be updated. The rules are [13]

$$\tau_{ij}(t+1) = (1-\rho)\tau_{ij}(t) + \sum_{s=1}^m \Delta\tau_{ij}^s(t),$$

$$\Delta\tau_{ij}^s(t) = \begin{cases} \frac{Q}{\cos t(i, j)}, & \text{ant } s \text{ in } [t, t+1], \\ 0, & \text{otherwise,} \end{cases} \quad (9)$$

where ρ is the information element volatilization coefficient and Q is the global information element constant; since the model target is the smallest cost, $\cos t(i, j)$ is the cost on the path (i, j) .

3.2. Firefly Algorithm. The firefly algorithm (FA) will find every feasible solution of the space and then perform a global search. The process of iterative search will be the attraction and movement process between individual fireflies [33, 34]. The basic principle involves two main factors: the luminous brightness of individuals in the population and the mutual

attraction between individuals. Luminous brightness is related to the fitness of an individual [35]. The better the fitness, the stronger the brightness; the attractiveness has a positive correlation with the luminous brightness and negative correlation with the distance between individuals; the brighter fireflies attract other individuals in the population toward their search area of better position movement, in order to complete the update of the firefly location individually over the whole process.

The luminous intensity I_i of individual i is represented by the fitness value, and the fitness function is generally the objective function, expressed as

$$I_i = f(X_i), \quad X_i = (x_{i1}, x_{i2}, \dots, x_{id}). \quad (10)$$

If the brightness of individual i is greater than that of individual j , the attraction of firefly i to firefly j , $\beta_{i,j}$ varies with distance

$$\beta_{i,j} = \beta_0 \exp(-\gamma r_{i,j}^2). \quad (11)$$

In the formula, β_0 and γ are the maximum absorption and the light absorption coefficient, respectively. $\gamma \in [0.01, 100]$; $\gamma_{i,j}$ is the Euclidean distance between individual i and individual j .

Individual j is attracted to individual i , and the position update formula is

$$X_j(t+1) = X_j(t) + \beta_{i,j}(X_i(t) - X_j(t)) + \alpha, \quad (12)$$

where α is a random control parameter, $\alpha \in [-1, 1]$.

3.3. Firefly-Ant Colony Hybrid Algorithm. The ant colony algorithm does well in the VRPSTW problem [36]. Because of its strong information feedback mechanism, it has a high solution efficiency for large-scale problems, but it is easy to produce pheromone when the pheromone concentration is too high in the later stage of the iteration. The stagnation leads to premature convergence and thus falls into the local optimum. In response to this problem, consider the self-organizing movement of the inferior individuals to the superior individuals in the firefly optimization, so that the population can complete the iterative update of individual positions during the process of gathering to the superior individuals, and new solution spaces can be searched in this process. Based on this, it is proposed to introduce the above optimization process between individual fireflies into ant colony optimization [37]. Taking advantage of these two methods, a firefly-ant colony hybrid algorithm (ACO_FA) is proposed, in order to generate additional satisfactory solutions in each iteration for ant optimization. Pheromone perturbation is to avoid the stagnation of pheromone, accelerate the premature convergence of the algorithm, and avoid stopping at the local optimal. With the ant colony algorithm as the basic framework, the optimization process of the individual firefly is used to expand the search solution space, and the iterative process is controlled. The aggregation of pheromone in the medium proposes a pheromone exchange process (hereinafter referred to as the FA search process). The algorithm flow is shown in Figure 1.

3.4. Particle Filter. The particle filter is a promotion of the Kalman filter and is a very novel tracking model. In 1992, Godon officially launched the model in his paper [38], and its significance is to use the Monte Carlo experiment method to solve the nonlinear filter Problem. However, because the resampling problem was not solved at that time and due to the limitations of the hardware conditions at the time, it was not practically applied. In recent years, as the resampling problem is solved, its applications have gradually increased, especially for nonlinear and non-Gaussian distribution. In short, the main idea is to use a certain number of particles $\{x^i, \omega^i\}$ to represent the probability density of the state. According to the Bayesian inference method, we get the posterior probability, filter out particles with high probability, discard small particles, and then obtain the state estimate of the system by a weighted average [39].

In the Bayesian filter scheme, for the known observation sequence $Z_1: t = \{Z_1, \dots, Z_t\}$, the following method can be used to iteratively update the posterior distribution of the state X_t :

$$\begin{aligned} P(X_t | Z_{1:t}) &= kP(Z_t | X_t)P(X_t | Z_{1:t-1}) \\ &= kP(Z_t | X_t) \int_{x_{t-1}} P(X_t | Z_{t-1})P(X_{t-1} | Z_{1:t-1}). \end{aligned} \quad (13)$$

Here, the conditional probability density $P(Z_t | X_t)$ is used to represent the observation model, and $P(X_t | X_{t-1})$ is used to represent the system model. In the particle filter model, N weighted particles $\{X(i)t-1, \omega(i)t-1\} Ni=1$ iteratively approximate the posterior distribution $P(X_{t-1} | Z_{1:t-1})$, where $\omega(i)t-1$ is the weight of the particle $X(i)t-1$. In this way, approximate integration by the Monte Carlo filter method can be obtained:

$$P(X_t | Z_{1:t}) \approx kP(Z_t | X_t) \sum_i \omega_{r-1}^{(i)} P(X_t | X_{r-1}^{(i)}). \quad (14)$$

The particle filter can be regarded as a priority sampler based on this probability. In particular, if N particles X_t^i are obtained from the proposed distribution, namely,

$$q(X_t) = \sum_i \omega_{r-1}^{(i)} P(X_t | X_{r-1}^{(i)}), \quad (15)$$

then priority can be expressed as

$$\omega_t^{(i)} = P(Z_t | X_t^{(i)}). \quad (16)$$

From this, we can get the posterior distribution $P(X_t | Z_{1:t})$ represented by the weighted particle $\{X(i)t \text{ at time } t, \omega(i)t\} Ni=1$. The procedure of our proposal can be described as follows:

- (1) State prediction: the next state is predicted by the system model, such as the AR model.
- (2) Calculate priority: calculate $P(Z_t | X_t)$ according to the matching degree of the target in the target search window.

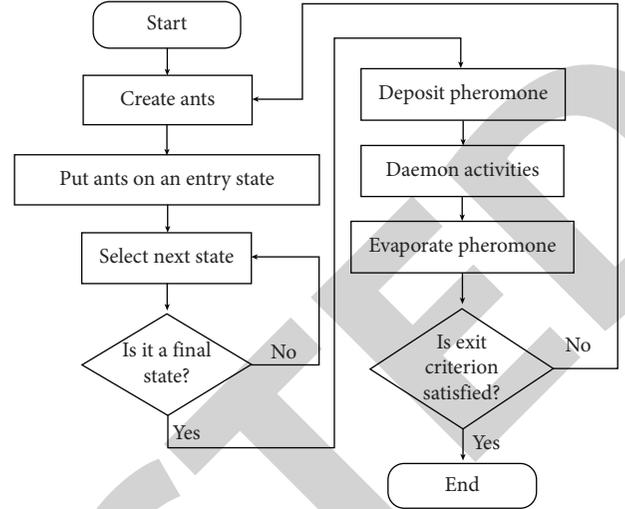


FIGURE 1: Flowchart based on genetic algorithm search.

- (3) Resampling: particle degradation should be discarded and the particles are “resampled” according to the priority value. The same number of particles is obtained. The result is that the particles with the higher priority are amplified, and the particles with the lower priority are suppressed.
- (4) Normalization: calculate the average value of the priority of all particles to get the normalized weight.

4. Overall Algorithm for the Problem

The algorithm first uses particle filters for visual tracking and then obtains the current operating environment of the vehicle; then, it performs cluster analysis on customer locations and finally performs path recognition in each area. The algorithm not only introduces particle filters, which are advanced visual tracking. The algorithm also improves the cuckoo search algorithm, when the bird’s egg is found by the bird’s nest owner; it needs to randomly change the position of the entire bird’s nest, which speeds up the search speed of the optimal delivery route. The detail of the method is presented in Figure 2.

5. Experiments

5.1. Experimental Setup. In order to improve the hybrid ant colony algorithm, with the goal of minimizing the total cost of delivery under the soft time window condition, the program is written by python software, and under the same parameter settings and experimental conditions, two methods are proposed in the article. The algorithm performs 500 iterations and runs the program 10 times for the given numerical experiment and the results can be shown in Table 2.

The OS, AS, AD, MD, and ACT in Table 2 represent the optimal solution, average solution, average deviation, maximum deviation, and average calculation time, respectively. It can be seen from Table 3 that the hybrid algorithm has significantly improved the accuracy of the solution. The global optimal solution is optimized by about 15.4% and the global

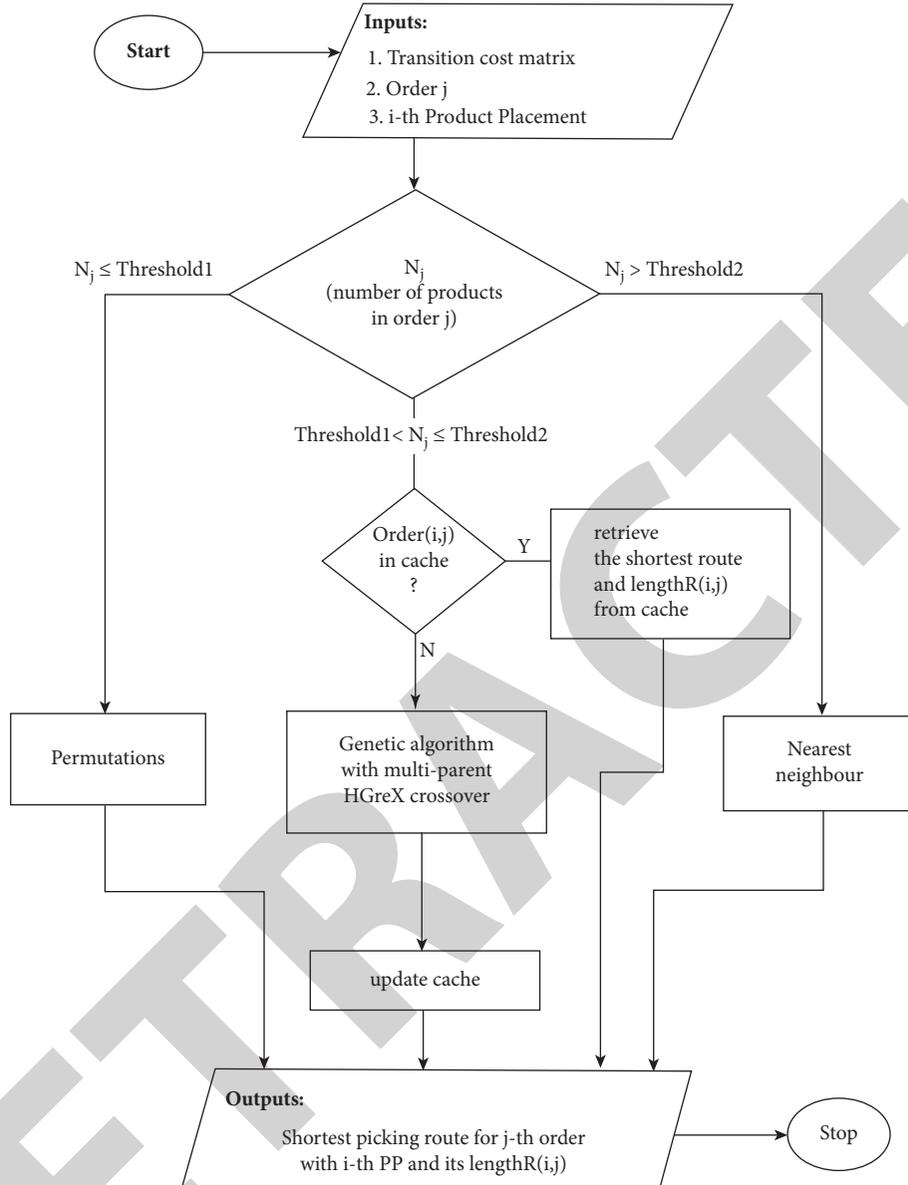


FIGURE 2: Flowchart of the proposed method.

average solution is optimized by about 15.6%, reflecting the algorithm. In the optimization process, the disadvantage of ant pheromone stagnation is solved to jump out of a local optimal state; secondly, the average deviation of the solution is reduced by about 0.21% and the maximum deviation is reduced by about 0.48%, indicating that the algorithm is improving the optimal state. In addition to the solution quality, the robustness of the algorithm is also improved, showing good solution performance. Since the proposed algorithm adds the FA search process on the basis of the original ant colony algorithm, it needs to spend extra time in the field search, so more time is needed in the calculation.

Obviously, this model is effective. Under the condition of limited vehicle load, the distribution center tries to meet the customer's time window as much as possible, while keeping the total cost small. The minimum distribution cost is 3342 yuan, and the distribution route 1 is 1-14-2-16-4-5-9-3-1

TABLE 2: Statistical results of numerical experiments.

Algorithm	OS	AS	AD	MD	ACT
ACO	2556	3045	3.86	9.56	19.027
The proposed method	5262	3489	9.05	8.08	20.016

and the vehicle full load rate is 100%; the delivery route 2 is 1-7-6-10-18-8-1 and the vehicle full load rate is 95%; the delivery route 3 is 1-15-11-12-17-13-1 and the vehicle full load rate is 88.33%; and the distribution route 4 is 1-20-19-1 and the vehicle full load rate is 43.33%.

Since the basic principle of the eye tracker we designed is to use the particle filter model to track the eyes. The state takes the position of the center of the target window, and the initial state predicts the position of the next frame according to the system model; that is, a certain number of particles are generated by statistical methods. And then, calculate the

TABLE 3: Related parameters of the proposed algorithm.

Parameter	Meaning	Parameter value
m	Number of populations	31
iter_max	Maximum number of iterations	500
Q	Global pheromone constant	10
α	Pheromone importance factor	1
β	Heuristic function importance factor	5
ρ	Pheromone volatilization coefficient	0.2
γ	Attraction control parameter (FA)	0.05
ε	Random control parameter (FA)	1

posterior probability by the observation model. Here, the probability density on the optimal path is used to measure the likelihood of the eyes. Finally, resampling filters the particles, amplifies the particles with high priority. For particles with low priority, the whole process is repeated iteratively. When the tracking fails, the entire image needs to be rescanned, and then the tracking is performed according to the above method.

The experimental data of the calculation example is randomly generated by Python into a distribution center and 19 customer sites. All customer points are randomly distributed in the plane coordinates of $(0, 70)2$; customer demand is an integer (t) randomly generated in the interval $[0, 20]$; time window and other information are shown in Table 2; vehicle speed $v=2$ km/min; the weight of a single vehicle is 60t; the service time per unit demand of the customer is 1 min/t; and the unit distance cost is 5 yuan/km. Table 4 shows the relevant parameters. Related parameters of the proposed algorithm are shown in Table 3.

5.2. Model Calculation Results and Analysis. For testing the reliability of our proposal, we recorded a 50-frame video sequence. In this sequence, the human face rotates in different directions and has rapid eye-closing behavior. Strong lighting interference: under this condition, the effect of eye tracking is shown in Figure 3. The scale of the image is 160×120 , and the number of particles is selected as $N=500$. It can be seen that, for different poses, and different eye opening, the algorithm proposed in this article can track the eyes well. However, the traditional brightness-based algorithm, under the above conditions, cannot track the eyes due to insufficient brightness. Figure 3 shows the predicted value of the eye position state and the state of the 50 frames. The filter value and the use of resampling technology prevent the degradation of particles, indicating that the particle filter model can track the target well.

It indicated that the accuracy used in this paper is higher than that of the Kalman filter, which is mainly related to the eye movement trajectory. In the sequence used in this paper, the eye movement trajectory is a nonlinear curve, so the Kalman filtering accuracy is not high. The algorithm in this paper can indeed handle the problem of nonlinear tracking. Meanwhile, experiments show that the algorithm can converge very quickly. This is mainly due to the fact that the

TABLE 4: Customer information form.

ID	Demand (t)	Service time (min)	Time span (min)
1	0	0	
2	5	10	[100, 150]
3	6	12	[85, 110]
4	4	8	[50, 60]
5	10	20	[40, 60]
6	18	36	[104, 190]
7	14	28	[90, 120]
8	12	24	[40, 55]
9	8	16	[70, 90]
10	9	18	[150, 175]
11	3	6	[220, 250]
12	17	34	[180, 190]
13	1+	32	[78, 85]
14	20	40	[256, 280]
15	11	22	[300, 340]
16	7	14	[310, 350]
17	6	12	[420, 460]
18	4	8	[450, 490]
19	16	32	[310, 330]
20	10	20	[510-600]

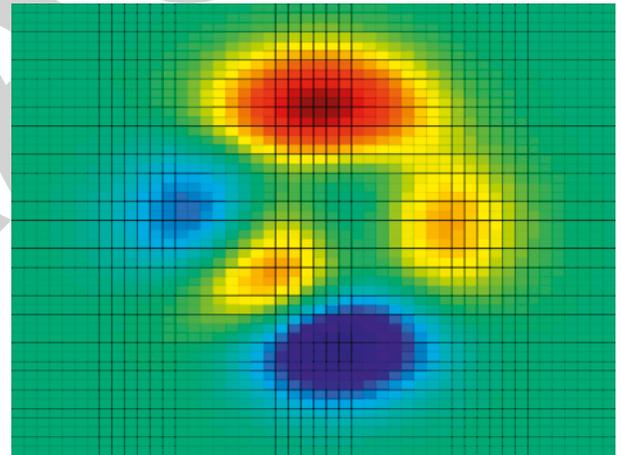


FIGURE 3: Probability distribution of predicted value and filtered value of the binocular position.

algorithm uses a sample set to represent the distribution of states and evolves particles in an optimized way, which shows great flexibility; that is, particles with high priority are considered, while particles with low priority are taken into account. We compared the mean square error values of the two algorithms: $MSEPF = 8.0953$ and $MSEKF = 17.8367$. The particle filter algorithm is also outperformed compared to the Kalman filter one. Figure 2 is the effect diagram of the eye tracking of the system, which shows that our proposal can supervise the goal in real time. When the number of particles tends to infinity, the particle filter can guarantee to converge to the approximate true value. Therefore, theoretically speaking, when the number of particles tends to infinity, the distribution can converge to the true value. However, there are no mature methods and theoretical guarantees on how to determine the lower limit of the number of particles that meet a certain accuracy. The number of particles used for

state estimation is larger, which can ensure accuracy, but real-time performance will be decreased. A compromise is needed between accuracy and real-time performance. In addition, resampling technology is also a key issue. The probability distribution of the predicted value and filtered value of the binocular position is shown in Figure 4.

In order to investigate the effectiveness of our proposal and other methods. Here, we take the F1-score into account to assess the experiment results, which can be defined as follows. And we also compared three methods proposed in the lasted three years such as GNB, RNB, and CNB with our proposal to investigate the effectiveness of our methods. And all these methods are introduced in Section 3. The results can be shown in Figure 5. As shown in Figure 5, the dark green bar represents our proposal and on all path situations, we can see that our proposal is better than others except for path 25, in which all methods obtain the same results. It indicated that our proposal can perform better than the other three methods. For more detail, we use the confuse matrix, accuracy, and precision to evaluate the model performance; the matrix can be defined as follows: (Table 5)

$$\begin{aligned}
 \text{accuracy} &= \frac{TP + TN}{TP + TN + FP + FN}, \\
 f1 - \text{score} &= \frac{2 * \text{precise} * \text{recall}}{\text{precise} + \text{recall}}, \\
 \text{precise} &= \frac{TP}{TP + FP}, \\
 \text{recall} &= \frac{TP}{TP + FN}.
 \end{aligned} \tag{17}$$

The results can be shown in Figure 5; the values of the red curve represent the results of our proposal and the others can be denoted as notations in Figure 5. Here, we compared four optimization algorithm based methods and three network based methods to investigate their prediction accuracy. The results show that our proposal GA-BASED outperforms other methods in all experimental cases and the other optimization methods can do better than most network based methods like GNB CNB and RNB. And we can see that our proposal is better than others significantly when the number of paths is 25, where all methods have the same results, which indicated that our proposal can perform better than the other three methods. As Figure 5(b) shows, except for the PSO-based method, the other methods including our proposal have the same time consumption, which indicated they can have the same time complication. From both figures, we can conduct that the GA-based method can do the best in all methods.

Comparative analysis of quality before and after optimization is given in Table 6. The minimum mass of the PATHS is the objective function, the thickness of the upper

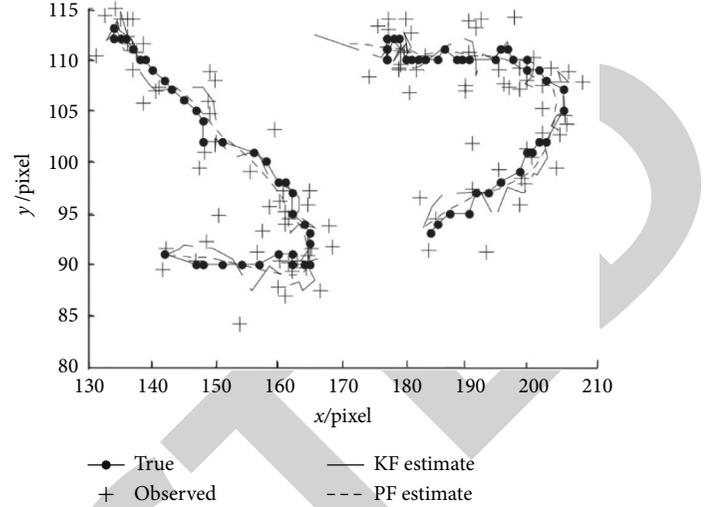


FIGURE 4: Probability distribution of predicted value and filtered value of the binocular position.

and lower plate is the design variable, and the stress intensity of the upper and lower plate is less than 340 as the constraint condition to optimize the size. After optimization, the safety factor is more than 1.4 except for the braking condition, and the safety factor is still above 2 in the other conditions. Under the condition of not changing the structure of the control arm, the research and design of the available dimension optimization can quickly reduce the quality and guide the development and design of the path identifications.

According to the calculation results, the overall fatigue life of the method is very good. However, the fatigue life in the partial stress concentration region is not very high, which is lower than the number of $10e6$ cycles, such as the back bushing joint, the front bushing joint, the reinforcement arm joint, and the outer end joint. These areas are prone to fatigue failure due to the large structural changes and obvious stress concentration. The service life of the control arm of the car suspension is given in Figure 6. We can see that the proposed algorithm can have the longest lifetime. This is because the proposed algorithm can cope with more difficult situations by adopting the neural network control method. Service life of the control arm of the car suspension is shown in Figure 7.

The result of fatigue simulation is the number of cycles, and it is necessary to convert the number of cycles into mileage. According to the national standard, the speed of the calculation is 60 km/h, the time represented by each cycle number is 60 s (the input time of Gen 11 road spectrum), the shortest cycle number of the control arm is 41680 times of the fatigue life of unit 180748. The equivalent road mileage can be expressed as 41680 km and the minimum fatigue life of the control arm is 41680 km, which meets the reliability test requirements of the general automobile of 3000 km.

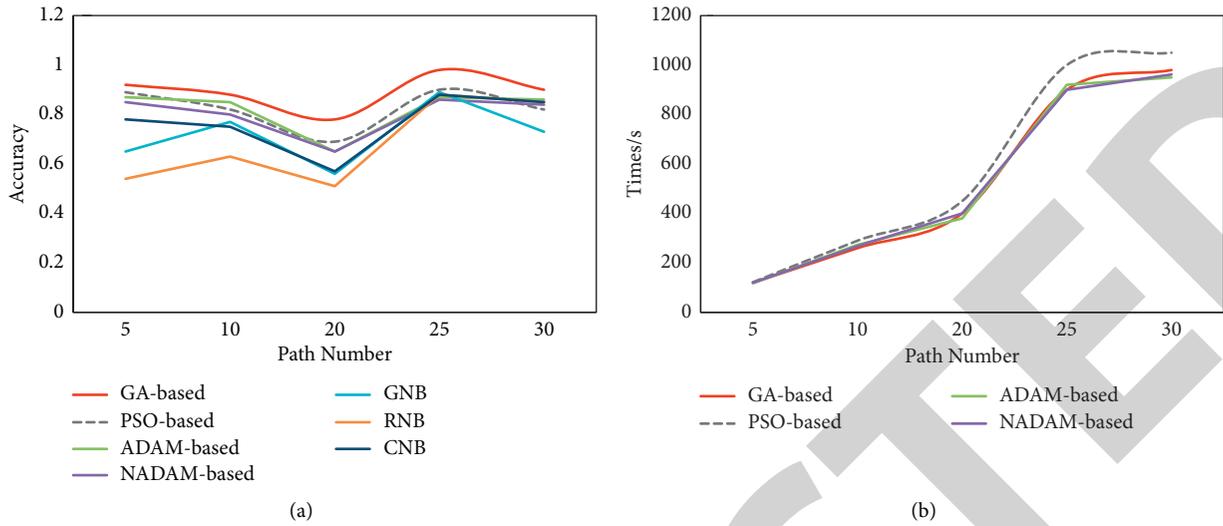


FIGURE 5: The results of all methods on path recognition. (a) Accuracy. (b) Time consumption.

TABLE 5: Matrix definition value list.

Real prediction	Positive	Negative
Positive	TP	FN
Negative	FP	TN

TABLE 6: Comparative analysis of quality before and after optimization.

	Preoptimization quality (kg)	Optimized quality (kg)	Reduce quality (kg)	Percentage (%)
On board	2.703	1.923	0.78	28.86
Lower plate	1.615	1.293	0.322	19.94

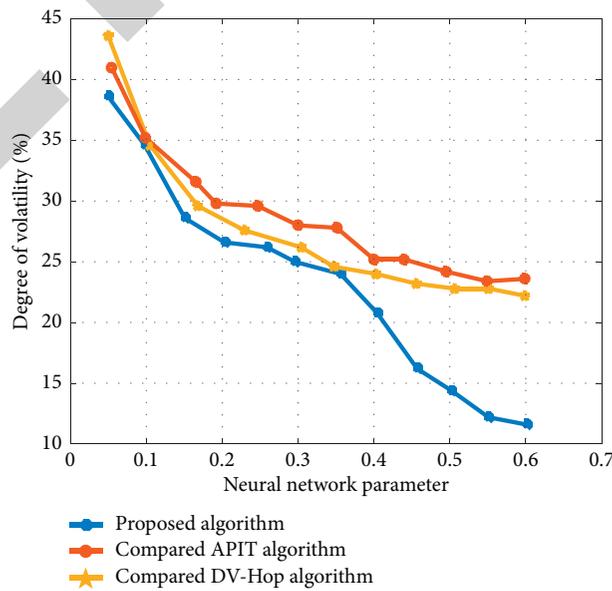


FIGURE 6: Vehicle stability analysis.

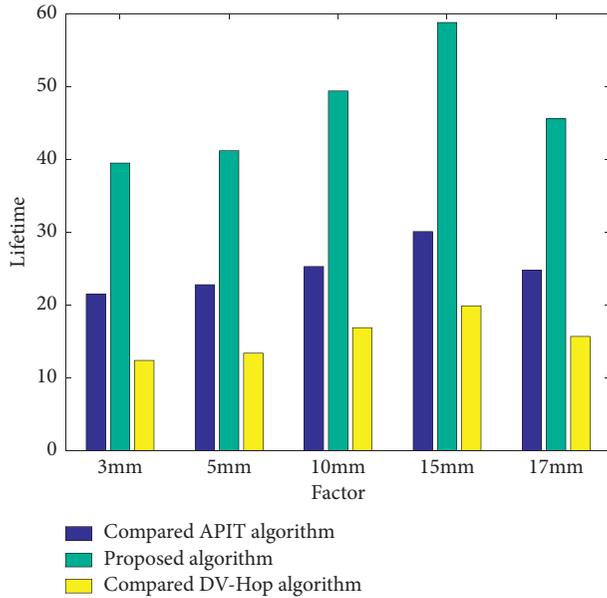


FIGURE 7: Service life of the control arm of the car suspension.

6. Conclusion

Taking into account the pheromone stagnation problem of the ant colony algorithm in the iterative process, the optimization mechanism of the firefly in the firefly algorithm is introduced into the ant colony algorithm to purposefully expand the search solution space and maintain. The solution of each generation has diversity, in order to optimize the update mechanism of the ant pheromone concentration, overcome the bottleneck of converging the local optimum, finally improve the accuracy of the solution, and mostly maintain the efficiency of the original algorithm. Besides, we employ the particle filter algorithm based on particle probability density. The algorithm uses the priority of the particle set to represent the distribution of the state. According to the Bayes theorem, the posterior distribution of the particles is obtained, and then the state is estimated after resampling. Since this algorithm is based on statistical experiments on the particle set, it has a wider range of applications than the Kalman algorithm. At the same time, the resampling method has been improved. The particle set model can better represent the probability distribution so that the model can be applied to nonlinear, non-Gaussian distribution occasions, and this is sometimes very important in reality. Experiments show that the method has good robustness in the intelligent recognition method of a vehicle path. Although our method has achieved good prediction accuracy at present compared with other popular methods, it is still unable to achieve considerable accuracy in the face of complex paths identification environment, and the training time of the model is long, which cannot meet the purpose of real-time prediction. In the future, we will further optimize the model to improve the training speed while ensuring accuracy.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The author declares that he has no conflicts of interest.

Acknowledgments

This article is one of the phased achievements of the 2020 Yulin Science and Technology Plan Project “The Construction of Yulin Software Engineering Technology Research Center-Modern Optimization Algorithms and Their Application in Optimal PATH Selection” (Project no. CXY-2020-007).

References

- [1] A. Alihodzic, E. Tuba, R. Capor-Hrosik et al., “Unmanned aerial vehicle path planning problem by adjusted elephant herding optimization,” in *Proceedings of the 2017 25th Telecommunication Forum (Telfor)*, pp. 1–4, IEEE, Belgrade, Serbia, November 2017.
- [2] K. Okamoto and P. Tsiotras, “Optimal stochastic vehicle path planning using covariance steering,” *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2276–2281, 2019.
- [3] G. Qing, Z. Zheng, and X. Yue, “Path-planning of automated guided vehicle based on improved Dijkstra algorithm,” in *Proceedings of the 2017 29th Chinese control and decision conference (CCDC)*, pp. 7138–7143, IEEE, Chongqing, China, May 2017.
- [4] P. X. Zhao, W. H. Luo, and X. Han, “Time-dependent and bi-objective vehicle routing problem with time windows,” *Advances in Production Engineering & Management*, vol. 14, no. 2, pp. 201–212, 2019.
- [5] H. Fan, Y. Zhang, P. Tian, Y. Lv, and H. Fan, “Time-dependent multi-depot green vehicle routing problem with time windows considering temporal-spatial distance,” *Computers & Operations Research*, vol. 129, Article ID 105211, 2021.
- [6] Y. Guo, S. Yan, J. Gong, L. Jin, and B. Shi, “The protective effect of selenium on bovine mammary epithelial cell injury caused by depression of tr,” *Biological Trace Element Research*, vol. 184, no. 1, pp. 75–82, 2018.
- [7] Z. Zhang, K. Wang, L. Zhu, and Y. Wang, “A Pareto improved artificial fish swarm algorithm for solving a multi-objective fuzzy disassembly line balancing problem,” *Expert Systems with Applications*, vol. 86, pp. 165–176, 2017.
- [8] R. W. White, S. T. Dumais, and J. Teevan, “Characterizing the influence of domain expertise on web search behavior,” in *Proceedings of the 2nd ACM international conference on web search and data mining*, pp. 132–141, Barcelona Spain, February 2009.
- [9] G. Ochoa, M. Hyde, T. Curtois et al., “Hyflex: a benchmark framework for cross-domain heuristic search,” in *Proceedings of the European Conference on Evolutionary Computation in Combinatorial Optimization*, pp. 136–147, Springer, Málaga Spain, April 2012.
- [10] J. Rubinovitz, J. Bukchin, and E. Lenz, “Ralb—a heuristic algorithm for design and balancing of robotic assembly lines,” *CIRP annals*, vol. 42, no. 1, pp. 497–500, 1993.

- [11] T. Logenthiran, D. Srinivasan, and T. Z. Shun, "Demand side management in smart grid using heuristic optimization," *IEEE transactions on smart grid*, vol. 3, no. 3, pp. 1244–1252, 2012.
- [12] S. Batistić and P. Van Der Laken, "History, evolution and future of big data and analytics: a bibliometric analysis of its relationship to performance in organizations," *British Journal of Management*, vol. 30, no. 2, pp. 229–251, 2019.
- [13] X.-S. Yang and A. Slowik, "Firefly algorithm," *Swarm Intelligence Algorithms*, CRC Press, Boca Raton, FL, USA, 2020.
- [14] Y. Hua and S. Juncheng, "Impact of three different habitats on photosynthetic physiology and conductivity of viburnum sargentii," *Journal of Northeast Forestry University*, vol. 8, 2013.
- [15] J. Tang, Z. Pan, R. Y. K. Fung, and H. Lau, "Vehicle routing problem with fuzzy time windows," *Fuzzy Sets and Systems*, vol. 160, no. 5, pp. 683–695, 2009.
- [16] T. Mouktonglang and P. Worapun, "A comparison of robust criteria for vehicle routing problem with soft time windows," *International Journal of Mathematics and Mathematical Sciences*, vol. 2019, Article ID 9137458, 7 pages, 2019.
- [17] Y.-H. Lin, R. Batta, P. A. Rogerson, A. Blatt, and M. Flanigan, "A logistics model for emergency supply of critical items in the aftermath of a disaster," *Socio-Economic Planning Sciences*, vol. 45, no. 4, pp. 132–145, 2011.
- [18] J. Xu, F. Yan, and S. Li, "Vehicle routing optimization with soft time windows in a fuzzy random environment," *Transportation Research Part E: Logistics and Transportation Review*, vol. 47, no. 6, pp. 1075–1091, 2011.
- [19] C. Lin, K. L. Choy, G. T. S. Ho, S. H. Chung, and H. Y. Lam, "Survey of green vehicle routing problem: past and future trends," *Expert Systems with Applications*, vol. 41, no. 4, pp. 1118–1138, 2014.
- [20] S. Yang, Z. Gong, K. Ye, Y. Wei, Z. Huang, and Z. Huang, "EdgeRNN: a compact speech recognition network with spatio-temporal features for edge computing," *IEEE Access*, vol. 8, pp. 81468–81478, 2020.
- [21] J. Zhang and D. Tao, "Empowering things with intelligence: a survey of the progress, challenges, and opportunities in artificial intelligence of things," *IEEE Internet of Things Journal*, vol. 8, no. 10, pp. 7789–7817, 2021.
- [22] M. S. Hossain and G. Muhammad, "An audio-visual emotion recognition system using deep learning fusion for a cognitive wireless framework," *IEEE Wireless Communications*, vol. 26, no. 3, pp. 62–68, June 2019.
- [23] A. V. Donati, R. Montemanni, N. Casagrande, A. E. Rizzoli, and L. M. Gambardella, "Time dependent vehicle routing problem with a multi ant colony system," *European Journal of Operational Research*, vol. 185, no. 3, pp. 1174–1191, 2008.
- [24] D. Sariklis and S. Powell, "A heuristic method for the open vehicle routing problem," *Journal of the Operational Research Society*, vol. 51, no. 5, pp. 564–573, 2000.
- [25] P. Francis, K. Smilowitz, and M. Tzur, "The period vehicle routing problem with service choice," *Transportation Science*, vol. 40, no. 4, pp. 439–454, 2006.
- [26] R. Montemanni, L. M. Gambardella, A. E. Rizzoli, and A. V. Donati, "Ant colony system for a dynamic vehicle routing problem," *Journal of Combinatorial Optimization*, vol. 10, no. 4, pp. 327–343, 2005.
- [27] P. Stodola and J. Mazal, "Applying the ant colony optimization algorithm to the capacitated multi-depot vehicle routing problem," *International Journal of Bio-Inspired Computation*, vol. 8, no. 4, pp. 228–233, 2016.
- [28] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.
- [29] S. Mazzeo and I. Loiseau, "An ant colony algorithm for the capacitated vehicle routing," *Electronic Notes in Discrete Mathematics*, vol. 18, pp. 181–186, 2004.
- [30] R. S. Parpinelli, H. S. Lopes, and A. A. Freitas, "Data mining with an ant colony optimization algorithm," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 4, pp. 321–332, 2002.
- [31] V. Maniezzo, L. M. Gambardella, and F. De Luigi, "Ant colony optimization," *New Optimization Techniques in Engineering*, Springer, Berlin, Germany, 2004.
- [32] L. P. Ding, Y. X. Feng, J. R. Tan, and Y. C. Gao, "A new multi-objective ant colony algorithm for solving the disassembly line balancing problem," *International Journal of Advanced Manufacturing Technology*, vol. 48, no. 5-8, pp. 761–771, 2010.
- [33] X. S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *International Journal of Bio-Inspired Computation*, vol. 2, no. 2, pp. 78–84, 2010.
- [34] S. M. Farahani, A. A. Abshouri, B. Nasiri, and M. R. Meybodi, "A Gaussian firefly algorithm," *International Journal of Machine Learning and Computing*, vol. 1, no. 5, pp. 448–453, 2011.
- [35] F. Zhang, "Research on improving prediction accuracy of sports performance by using glowworm algorithm to optimize neural network," *International Journal of Information and Education Technology*, vol. 9, no. 4, 2019.
- [36] D. H. Alman, M. E. Breton, and J. Barbour, "New results on the brightness matching of heterochromatic stimuli," *Journal of the Illuminating Engineering Society*, vol. 12, no. 4, pp. 268–274, 1983.
- [37] Y. Li, "Finite element structure analysis of automobile suspension control arm based on neural network control," *Security and Communication Networks*, vol. 2021, Article ID 9978701, 11 pages, 2021..
- [38] R. Fouladi, S. Salimi, and A. Salahi, "A graph based security dependency analysis of data communication networks to their topology," in *Proceedings of the 2009 4th International Conference on Computer Science & Education*, pp. 284–288, Nanning, China, July 2009.
- [39] K. Nummiaro, E. Koller-Meier, and L. Van Gool, "An adaptive color-based particle filter," *Image and Vision Computing*, vol. 21, no. 1, pp. 99–110, 2003.