Journal of Network Intelligence ©2024 ISSN 2414-8105
Taiwan Ubiquitous Information Volume **9**, Number **2**, May **2024**

Data Error Correction of WSN Nodes for Cold Chain Logistics Tracking

Jiwu Wang*

Chuzhou Applied Technology School, Chuzhou 239500, P. R. China
498800048@qq.com

Hao Li

Department of Engineering,
Krirk University, Bangkhen 10220, Thailand
007.lihao@163.com

*Corresponding author: Jiwu Wang
Received September 23, 2023, revised December 9, 2023, accepted February 29, 2024.

Abstract. *A variety of different types of sensors are required to be used in the cold chain logistics system based on wireless sensor network, so how to make better use of each sensor to serve the cold chain link is a key research issue in the related field. During the cold chain transport process, the special environment of cold chain logistics and the signal quality of the sensor nodes may lead to inaccurate data recording, so it is necessary to modify the erroneous data and deal with it in time. To address the above problems, this work proposes a cold chain logistics sensor node data error correction method. Firstly, in order to keep the data such as temperature and humidity within a strict fluctuation range, it is proposed to use Kalman filtering algorithm for error correction of these data. Second, in order to accurately locate the original latitude and longitude data to the transport road, it is proposed to use the Hidden Markov Prediction Model to correct the original trajectory data. At the same time, for the situation that the Hidden Markov Model is easy to fall into the local optimum during training, the Baum-Welch algorithm and the initial parameters of the Hidden Markov Prediction Model are optimised using the improved genetic algorithm, and the trajectory bias correction model with a higher degree of accuracy is formed. The simulation results show that the proposed improved genetic algorithm can converge to the global optimum, which can improve the convergence accuracy and speed. The proposed trajectory bias correction model has a greater improvement in road matching accuracy than the existing comparison algorithms.*
Keywords: cold chain traceability; WSN; genetic algorithm; dynamic variation; subpopulation migration; Hidden Markov Models

---

1. **Introduction.** With the continuous development of smart hardware devices, their prices are decreasing, so that smart devices can carry a variety of sensors. And the communication between a variety of devices drives the further development of Wireless Sensor Network (WSN) [1]. The access devices in WSN can obtain more network data, and the physical entities can provide more intelligent services to the network users through the related data [2], which becomes the main driving force for the development of the Internet of Things (IoT).

WSNs provide humans and computers with the ability to learn from and interact with billions of sensor devices, enabling the seamless integration of the cyber world with the physical world [3]. Internet users can access the services provided by these sensor devices at any time over the Internet, fundamentally changing human interaction with the world. WSNs have good applications and development in many fields, such as building smart cities through wireless sensor networks, which allow real-time monitoring of parking spaces [4], urban noise [5], traffic congestion [5], and street lighting [6] for more effective management. Smart homes, environmental monitoring, disaster prediction, connected cars, various wearable devices and smart healthcare are also practical application scenarios for WSNs. However, the rapid development of WSN is also accompanied by many challenges, such as the security of IoT [7]. How to build a fair and trustworthy WSN platform is a must think about. The node data error correction technology of WSN plays an important role in improving the reliability, accuracy, and energy efficiency of sensor data transmission, and it is one of the key technologies to build a WSN system for the following reasons:

(1) Data transmission reliability: In wireless sensor networks, sensor nodes usually lack strong signal transmission and reception capabilities. Due to factors such as weak signals and complex transmission medium environment, data transmission between nodes may be affected by interference, noise, attenuation, etc., resulting in data loss, damage or error. Error correction techniques can help to recover or repair these erroneous data and improve the reliability of data transmission [8].

(2) Energy efficiency: wireless sensor nodes usually have a limited energy supply, such as batteries [9]. Error correction techniques for node data can prevent nodes from repeatedly sending lost data, thus reducing energy consumption and extending the life of the node.

(3) Data quality and accuracy: data collected by sensor nodes are used for analysis and decision making, e.g., environmental monitoring, health monitoring, etc. If the node data is erroneous or incomplete, it may lead to wrong analysis results and decision-making [10]. Through error correction techniques, the quality and accuracy of node data can be improved to ensure the accuracy of subsequent data processing and applications.

This work starts from a food safety supply project to study WSN node-based data correction method for tracking the cold supply chain. Due to the special characteristics of cold chain logistics, the data information involved in cold chain logistics has very strict requirements, and the accuracy and real-time data need to be effectively guaranteed. However, the information collected by each sensor due to quality, signal and other problems may be biased and data redundancy, which is unacceptable for the cold chain link with strict data requirements. Therefore, this work filters redundant data and corrects errors in the data collected from sensor nodes in the WSN environment, so as to guarantee the quality and precision of the information gathered by the middleware.

1.1. **Related Work.** Cold chain logistics tracking requires high data accuracy and real-time performance, while various types of sensors may have low data accuracy and missing data due to quality signals and other problems, which is unacceptable for cold chain logistics. Therefore, the objective of this work is to study the methods of error correction for the data collected by the sensors used in the cold chain, which may be inaccurate or even wrong due to the quality and other reasons.

In the cold chain logistics tracking application scenario, the data of WSN can be mainly classified into two types [11]: temperature data and location temperature data. Error correction techniques for these two types of data mainly involve two aspects: error correction for temperature data and error correction for location data.

Firstly, temperature sensors may be affected by factors such as noise, interference and sensor errors when collecting temperature data, thus introducing certain errors. In order to improve the accuracy and reliability of temperature data, researchers have proposed a variety of error correction techniques, which mainly include [12]: 1) Statistical analysis-based methods, such as filtering and smoothing algorithms, which are used to reduce the noise and smooth the temperature data. 2) Model-based methods, such as Kalman filter and adaptive filter, which use the system model to predict and correct the data. Li et al. [13] proposed an error modelling and compensation based approach for improving the accuracy and reliability of temperature sensor data. By modelling and correcting the errors, accurate collection and transmission of temperature data was achieved. 3) Error control coding based methods such as Cyclic Redundancy Check (CRC) and Forward Error Correction (FEC) [14] were used to detect and correct errors in transmission. Kim et al. [15] investigated an error control scheme in a WSN based temperature monitoring system. The quality and reliability of temperature data was improved by introducing error detection and correction techniques.

Secondly, in WSN, accurate node positioning is the key to cold chain logistics tracking. However, WSN node position estimation may be affected by sensor errors, signal attenuation, signal multipath effect, etc., which leads to errors in positioning data. In order to reduce positioning errors to improve positioning accuracy and reliability, scholars have investigated a variety of positioning data error correction techniques, including: 1) Signal processing-based methods such as signal attenuation compensation, multipath effect suppression, and positioning error correction. Lv et al. [16] proposed a robust positioning scheme for node positioning in WSNs. By using signal strength fingerprinting and diversity of sensors, the scheme can effectively reduce multipath effects and localisation errors, and improve localisation accuracy and reliability. 2) Probabilistic and statistical based methods such as Hidden Markov Models (HMMs) and Particle Filters (PFs) are used for prediction and correction of localisation data. Hidden Markov Model (HMM) based WSN trajectory mapping and localisation is a common approach to infer node position trajectories by analysing a sequence of observations from sensor nodes. Alabadleh et al. [17] proposed an HMM based localisation algorithm using various constraints such as proximity to nodes distance, node motion speed and edge effects, to improve the localisation accuracy of nodes in WSNs. The robustness and accuracy of localisation is enhanced by fusing multiple constraints into the state transfer and observation model of the HMM.

In trajectory localisation prediction, Bayesian networks, the conditions under which neural networks operate are more restricted, whereas HMM is a probabilistic model with the advantages of high modelling flexibility, low computational complexity and high scalability. The results of the study by Kalkha et al. [18] show that HMM can effectively deal with a variety of uncertainties present in WSNs, including noise, signal attenuation, and occlusion, etc. Xiong et al. [19] proposed an unrestricted and efficient attribute-based encryption scheme for adaptive security in cloud-assisted Internet of things environment. Attribute-based Encryption(ABE) [20] technology is used in this scheme, which allows encrypted data to be associated with specific attributes, thus realizing flexible access control. This scheme makes use of the cooperative work of broadcast data transmission and cloud server [21], and realizes efficient attribute analysis and key distribution. HMM can provide quantification of uncertainty in WSN positioning results and trajectories by estimating the probability distribution of node locations through probabilistic reasoning. However, HMM also suffers from some problems such as lack of training accuracy and long time, and the lack of accuracy can also lead to a decrease in the prediction accuracy afterwards.

1.2. **Motivation and contribution.** In terms of HMM model optimisation, researchers have improved the accuracy and reliability of HMMs in the fields of trajectory mapping, speech recognition, and natural language processing by combining heuristic algorithms with HMM models, such as Genetic Algorithms (GA), Particle Swarm Optimization, PSO) and Ant Colony Optimization (ACO). Heuristic algorithms are able to find the near-optimal solution quickly through search and optimisation techniques, thus improving the accuracy and efficiency of HMM models. However, how to select and design appropriate heuristic algorithms with the characteristics of specific problems is still one of the challenges in future research.

In order to improve the accuracy of cold chain logistics tracking, this work proposes a WSN node data error correction method. The main innovations and contributions of this work include:

(1) According to the required characteristics of cold chain data, an algorithm based on Kalman filter is adopted to process smooth data such as temperature and humidity, thus smoothing the data curve and reducing the anomalies.

(2) In order to improve the convergence accuracy and speed, a GA algorithm based on polynomial variation and adaptive migration, called PAGA algorithm, is proposed and its effectiveness is verified by several standard functions.

(3) For such irregular and highly variable positioning data as latitude and longitude, it is proposed to combine the electronic map and HMM model for trajectory mapping and positioning, and for the situation that the HMM model tends to fall into a local optimum during training, the PAGA algorithm is used to optimise the Baum-Welch algorithm and the initial parameters.

2. **Kalman filter based error correction for smooth data.** The real data in cold chain environment such as temperature and humidity are smooth and less fluctuating, while such data are continuous and not regular. Conventional linear equations and other methods of speculation for regular data are not applicable. In order to solve this kind of problem, this work adopts the Kalman filter based algorithm to process the cold chain data according to its required characteristics.

Kalman filtering is suitable for estimating the optimal state of dynamic data consisting of random variables. Since the data contains noise which results in inaccurate values measured by the sensors, we need to estimate the true value of the state using Kalman filtering. The state vectors in Kalman filtering are equipped with dimensionality. In the cold chain, most of this smooth data is one-dimensional. It implies that numbers may be calculated from data like temperature and humidity. For instance, let's say the internal temperature of the cold chain transport is maintained at T, while the cold chain vehicle will encounter many situations that may lead to changes in the temperature inside the temperature compartment during its movement, such as weather, air exchange between the compartment and the outside world, etc., which is referred to as transport noise, and the sensor measurement of the temperature is also subject to bias, which is referred to as measurement noise.

Assume that the temperature at the moment $k-1$ is T_{k-1}, the optimal deviation is T_{oe}, the prediction uncertainty bias is T_{ue}, the sensor measurement is T_s, and the sensor measurement bias is T_{se}. According to the principle of linear superposition, the prediction bias T_{pe} of the temperature is obtained as:

$$T_{pe} = \sqrt{T_{oe}^2 + T_{ue}^2} \tag{1}$$

Obviously, the value of T_{pe} is larger than T_{oe}, which is caused by the inclusion of prediction uncertainty noise. Since the desired temperature setpoint is fixed in cold chain vehicles, it is first assumed that the temperature at moment k is equal to the temperature at moment $k-1$, at which time there exist two temperature values T_{k-1} and T_s at moment k, at which time the Kalman gain K_g is:

$$K_g = \frac{T_{pe}^2}{T_{pe}^2 + T_{se}^2} \tag{2}$$

From this, the temperature at moment k is estimated to be:

$$T_k = T_{k-1} + K_g \cdot (T_s - T_{k-1}) \tag{3}$$

After finding T_k, the optimal deviation T_{oe} at the time T_k is required:

$$T_{oe} = \sqrt{(1 - K_g) \cdot T_{pe}^2} \tag{4}$$

T_{oe} at the time of T_k is applied to solve for the temperature at the time of T_{k-1}, thus iterating with the value measured by the temperature sensor to eventually derive temperature data that is closer to the true value.

3. PAGA-HMM modelling.

3.1. **Representation of Hidden Markov Models.** As a well-established model for statistical analysis [22], HMM is a derivation of Markov chain to describe a Markov process with implicit positional parameters. Due to the advantages of simplicity of modelling and clarity of physical meaning, HMMs are now successfully used in areas such as speech recognition, text recognition and trajectory mapping.

In general, the events observed by the HMM do not have a one-to-one correspondence with the states, but are linked to the states through probability distributions. The HMM model has two parts [23], one is the visible part and the other is the hidden part. In a simple Markov chain, it can be seen as the part that is visible to the observer, while those states that are not directly observable are the hidden part. So, each state may have a certain probability distribution on the output.

For the HMM model, we set Q to be the set of all possible occurrences of hidden states, and V to be the set of all possible observations to states.

$$Q = \{q_1, q_2, \ldots, q_N\}, \quad V = \{v_1, v_2, \ldots, v_M\} \tag{5}$$

where N and M denote the number of all possible hidden states and the number of all possible observations, respectively.

For a sequence of length T, we set S to be the sequence of states and O to be the sequence of observations.

$$I = \{i_1, i_2, \ldots, i_T\}, \quad O = \{o_1, o_2, \ldots, o_T\} \tag{6}$$

The formula for the HMM model is:

$$\lambda = (\pi, A, B, N, M) \tag{7}$$

where π is the initial state distribution vector, A is the transfer matrix between states, and B is the observation probability matrix.

$$A = \{a_{ij}\}, \quad a_{ij} = P(y_{ti+1} = s_j \mid y_{ti} = s_i), \quad 1 \le i, j \le N \tag{8}$$

$$B = \{b_{ij}\}, \quad b_{ij} = P(x_{ii} = o_j \mid y_{ii} = s_i), \quad 1 \le i \le N, \quad 1 \le j \le M \tag{9}$$

$$\pi = \{\pi_i\}, \quad \pi_i = P(y_{t1} = s_i), \quad 1 \le i \le N \tag{10}$$

3.2. **HMM model parameter optimisation problem.** Given a sequence of observations $O = \{o_1, o_2, \ldots, o_T\}$, we need to continuously adjust the size of the parameters $\lambda = (\pi, A, B)$ of the HMM model in order to maximise the probability that the observed sequence occurs under the model with this parameter $P(O \mid \lambda)$. This type of problem is also an optimisation problem for the model parameters [24]. The Baum-Welch algorithm is mainly used to train the observation sequence. The final obtained λ can explain the observation sequence well. However, in reality, the training process is numerous and complicated, and consumes a long time. The transitive expectation of state S_i to state S_j is:

$$\xi_t(i,j) = P(q_t = S_i, q_{t+1} = S_j | O, \lambda), 1 \le t \le T-1 \tag{11}$$

The corresponding states at time t and $t+1$ are S_i and S_j, respectively.

$$\xi_i(i,j) = \frac{\beta_i(i) a_{ij} b_j(O_{i+1}) \eta_{i+1}(j)}{\sum_{i=1}^{N} \sum_{j=1}^{N} \beta_i(i) a_{ij} b_j(O_{i+1}) \eta_{i+1}(j)}, 1 \le t \le T-1 \tag{12}$$

where $\beta_i(i)$ and $\eta_{i+1}(j)$ are derived using the forward and backward algorithms, respectively.

$$\beta_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^{N} \beta_i(i) \beta_{ij}, 1 \le t \le T-1 \tag{13}$$

$$\eta_t(i) = \sum_{j=1}^{N} \eta_{t+1}(j) a_{ij} b_j(O_{t+1}), 1 \le t \le T-1, 1 \le i \le N \tag{14}$$

A new model parameter λ' is obtained from the derived $\xi_i(i,j)$ and the loop continues until the model converges.

3.3. **PAGA algorithm design.** As a heuristic algorithm that simulates the biological evolution mechanism imitating the nature, GA algorithm is able to automatically acquire information during the search process and adaptively control the search process to get the optimal solution [25].

GA algorithm is different from traditional ground intelligence algorithm, its process simulates the natural selection, and there is little chance of elimination for the individuals with a large fitness function, and after surviving, it generates a new individual through mutation. The crossover approach means that there is only one crossover point on the chromosome code, and the exchanged segments are random, and new individuals are formed after the exchange [26]. The schematic diagram of chromosome single point crossover is shown in Figure 1. Mutation is a genetic algorithm in which some genes in the coding strings of an individual chromosome are replaced with other alleles to form a new individual.

Since the GA algorithm uses a stochastic search method, its search process may take a long time to find the optimal solution, especially when faced with complex problems [27]. In addition, during the optimisation process, the GA algorithm may converge to the local optimal solution too early and fail to discover the global optimal solution. This is because the crossover and mutation operations between parent individuals during the stochastic search may cause the genetic algorithm to fall into the local optimal solution and fail to jump out. Therefore, this work improves the traditional GA algorithm from two aspects and proposes a GA algorithm based on polynomial variation and adaptive migration, called PAGA algorithm.

Firstly, the polynomial variation operator based on dynamic variation rate is constructed to adjust the variable values in the later stage of the algorithm, in order to overcome the problem that the algorithm cannot converge after many generations or

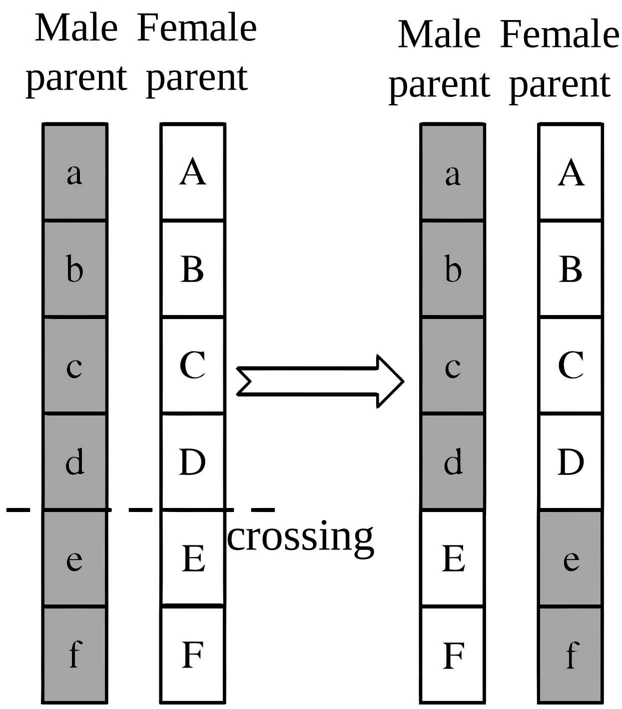


Figure 1. Schematic diagram of single point crossover of chromosomes

easily falls into the local optimum due to the small variation probability. Polynomial variational is a very effective and commonly used variational method [28] in the form of:

$$v_k' = v_k + \alpha \times (u_k - l_k) \tag{15}$$

$$\alpha = \begin{cases} [2u + (1-2u)\times \\ (1 - \frac{v_k - l_k}{u_k - l_k})^{\beta_m + 1}]^{\frac{1}{\beta_m + 1}} - 1, u \leq 0.5, \\ 1 - [2(1-u) + 2(u-0.5)\times \\ (1 - \frac{u_k - v_k}{u_k - l_k})^{\beta_m + 1}]^{\frac{1}{\beta_m + 1}}, \quad u > 0.5, \end{cases} \tag{16}$$

On this basis, a dynamic variability rate is introduced to adjust the probability values after each generation run as follows.

$$P_m(i+1) = P_m(i) + (\frac{g}{G})^n \times (\frac{P_c(i)}{3} - P_m(i)) \tag{17}$$

where u is a random number between $[0, 1]$, β_m is the user-specified distribution index, g is the current number of runs, G is the total number of runs, $P_m(i)$ is the probability of variation for the i-th population, and $P_c(i)$ is the probability of crossover for the i-th population.

Because the three key parameters of HMM matrix requires each row of probability sum of 1, the population after selection, crossover, mutation need to be normalised for each matrix, and to ensure that the maximum fitness of the population is only increasing rather than decreasing [29].

Secondly, an adaptive immigration operator using the hybrid subpopulation exchange mechanism is constructed. This operator increases the immigration between populations compared with the traditional operator, but the immigration with too large magnitude will cause the algorithm to fall into local optimum while improving the convergence speed, and the Hamming distance can measure the similarity between populations, so the type of immigration is selected according to the average weighted Hamming distance of individuals in the population [30], and the calculation method is shown as follows:

$$H_{\text{avg}} = \frac{1}{n}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\sum_{m=1}^{k}\omega(x_m)|b_{im} - b_{jm}| \tag{18}$$

$$\omega(x_m) = 2^m \tag{19}$$

The population migrant needs to find the maximum fitness value Fmax of each population and put it into the set F, and sort the populations according to the fitness value from the largest to the smallest, call the corresponding population with the largest value in F as the father's population and the corresponding population with the smallest value as the child's population. According to the size of the population's Hamming Distance, judge whether to replace the child's population with the father's population in order to increase the convergence speed. Individual immigration filters the best and worst individuals in each population based on fitness, and replaces the best individual of the former population with the worst individual of the latter population. To satisfy the closure, the best individual of the last population replaces the worst individual of the 1st population to achieve the information exchange between populations. The execution process of the adaptive migration operator is shown in Figure 2.

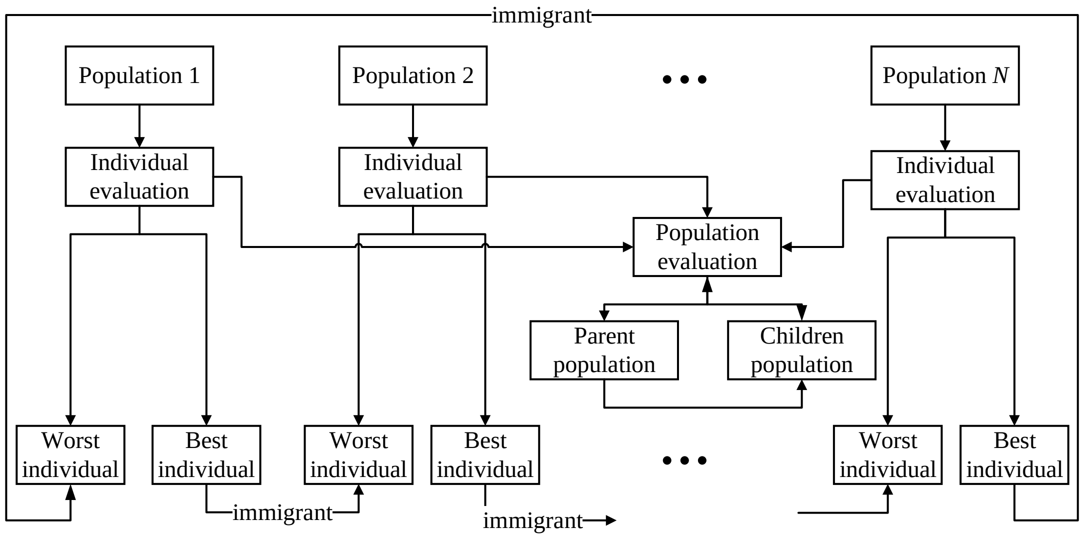


Figure 2. Execution process of the adaptive immigration operator

The criteria for selecting the type of immigrant are shown as follow :

$$\begin{cases} P_m, H_{avg}^1 - H_{avg}^2 \geq \xi \\ I_m, H_{avg}^1 - H_{avg}^2 < \xi \end{cases} \tag{20}$$

where P_m is the population immigration, I_m is the individual immigration, and ξ is the population Hamming distance threshold.

3.4. **Operational flow of the PAGA-HMM model.** Through the analysis, it can be seen that the process of training the HMM model by Baum-Welch algorithm needs to set the initialisation parameter (π, A, B), and iteratively obtain the new parameter so that the output probability $P(O \mid \lambda)$ is the most. However, the solution obtained by Baum-Welch algorithm is easily affected by the initial value and easily falls into the local optimum, while the traditional GA algorithm has the problem of immature convergence, in order to make up for the shortcomings of the two, this paper proposes the PAGA-HMM model, in which the selection of the important parameters is as follows:

(1) Selection of chromosome codes.

PAGA requires the encoding of the object it is seeking to optimise, and the solution process is represented using the encoding of the solution. Therefore, it is necessary to encode the initialisation parameters of the HMM model (π, A, B), and ensure that the sum of the elements of each row of (π, A, B) is 1. The specific form of the HMM chromosome encoding is shown in Figure 3, with the following constraints.

$$\begin{cases} \sum\limits_{i}^{N} \pi_i = 1 \\ \sum\limits_{j}^{M} a_{ij} = 1, 1 \leq i \leq N \\ \sum\limits_{j}^{M} b_{ij} = 1, 1 \leq i \leq N \end{cases} \tag{21}$$

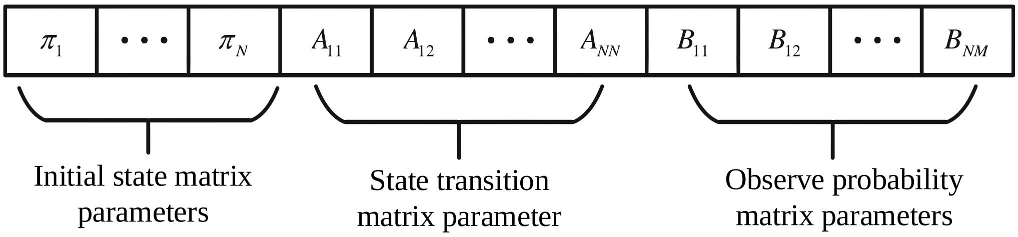


Figure 3. HMM chromosome coding

(2) Adaptation function.

The goodness of the model depends on the size of the fitness function value, the smaller the fitness function value the better the model. Considering that the aim of the HMM model when using the Baum-Welch algorithm is to find the parameter that maximises the conditional probability $P(O|\lambda)$, the fitness function chosen in this work is:

$$f_{fitness}(\lambda) = \log\left(P\left(O^k\right)|\lambda\right) \tag{22}$$

(3) Genetic operators.

The selection operator of the PAGA algorithm uses a roulette wheel approach, while the polynomial variation operator and the adaptive immigration operator are implemented as described above.

(4) Termination conditions.

A preset maximum number of evolutionary generations was used as a termination condition, and the maximum number of evolutionary generations G = 50 was set in this work.

Eventually, the operational flow of the PAGA-HMM model is shown in Figure 4.

4. **Trajectory correction based on the PAGA-HMM model.** The deskewing of vehicle trajectories [31] is the use of the PAGA-HMM model to solve its decoding problem. After knowing the latitude and longitude information acquired by the device, it goes to infer the real position information of the corresponding observation, to straighten out information that clearly doesn't fit with how a car moves on the road.

Firstly, we need to correspond the vehicle trajectory correction to the PAGA-HMM model, including two aspects.

(1) Visible Status Chain: latitude and longitude of the monitoring obtained from the positioning device.

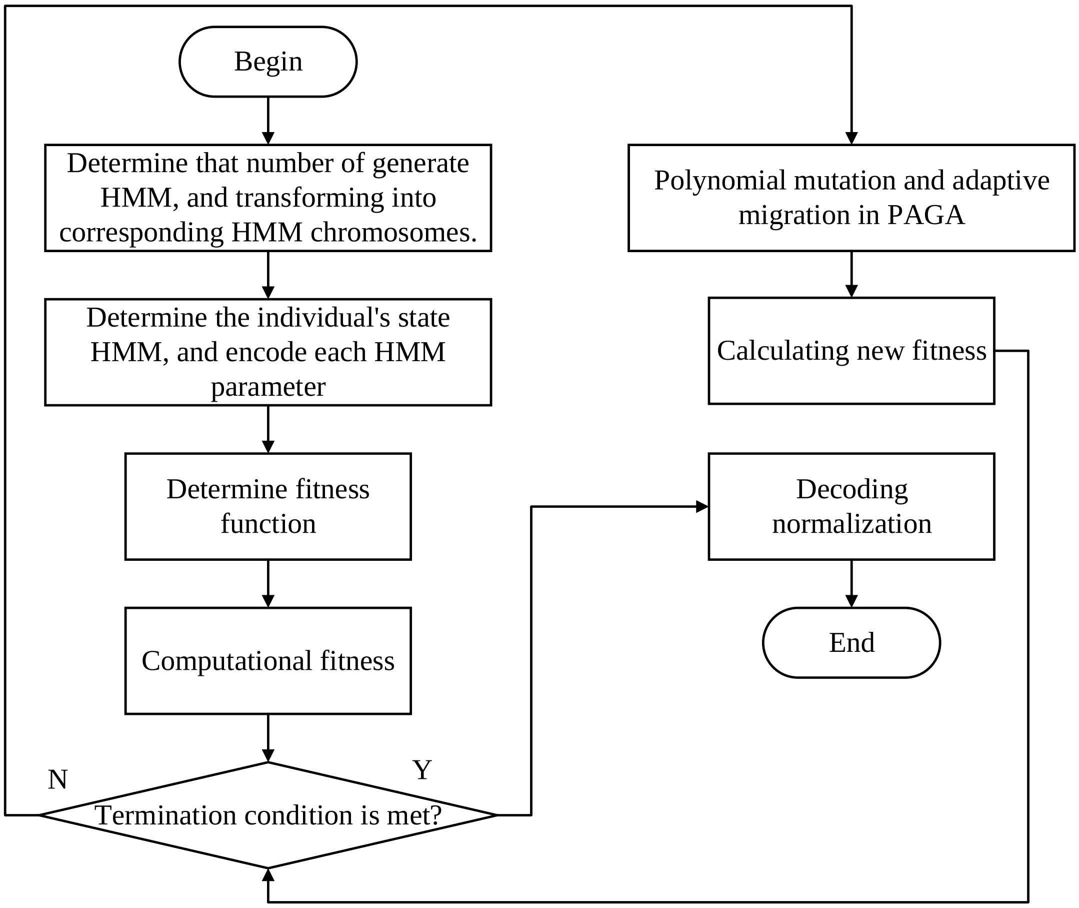


Figure 4. Run flow of the PAGA-HMM model

(2) Markov chain: obtaining the true location of the positioning device where the latitude and longitude information is located.

Create a sequence of positional information observations $Y = (Y_x|x = 1, 2, 3, \ldots, X)$, where X represents the number of observations (latitude and longitude). Create a road network observation sequence where $N = (V, A)$, $V = (p_i|i = 1, 2, 3, \ldots, N)$ represents the nodes on the roads and $A = (r_j|j = 1, 2, 3, \ldots, M)$ represents the roads on the road network in the track. Create a hidden sequence $R = (r_i|i = 1, 2, 3, \ldots, T)$ to locate the real position of the device.

When finding the state transfer probability of the vehicle deviation correction model, the relationship between the device positioning information should not be considered, but simply the data that exists in-between the actual locations of the candidates. Since the chance of a state transfer is proportional to the actual distance between the candidate points, the likelihood of a transfer increases as the distance between the candidate points decreases.

$$a_{ij} = \mathrm{p}(p_{x+1} = p_j|p_x = p_i) \propto \mathrm{e}^{-\beta d_{ij}} \tag{23}$$

where d_{ij} denotes the distance between the candidate points p_i and p_j, and β denotes the parameter that controls the element that affects the shortest distance between p_i and p_j.

After establishing the vehicle trajectory deviation correction model based on PAGA-HMM, the Viterbi algorithm [?] is used to solve it. The Viterbi algorithm, which essentially uses dynamic programming to solve the problem, is used to solve for the probabilistic maximum path in the trajectory deviation correction model. The iterative formula is as follows:

$$\delta_t(i) = \max_{i_1, i_2, ..., i_{t-1}} P(i_t = o_1, o_2, ..., o_t|\lambda), i = 1, 2, ..., N \tag{24}$$

5. Experimental results and analyses.

5.1. Validation of the PAGA algorithm. The simulation experiments were carried out in MATLABR2014a environment, with the operating system Windows7, CPU i3-4160, and memory 8 G. Firstly, the validation of the effectiveness of the PAGA algorithm was carried out. Then, the feasibility verification of the vehicle trajectory correction model based on PAGA-HMM was carried out.

In order to verify the performance of the proposed PAGA algorithm for merit seeking, four standard test functions were selected for merit seeking experiments and compared with standard GA, IGA [33], and MGA [34]. All GA algorithms have a population of size 50, an iteration number of 200, and the selection operation is roulette selection.

(1) Sphere function.

$$f_1(x) = \sum_{i=1}^{n} x_i^2, \quad -100 \le x_i \le 100, \quad \min f_1(x) = f_1(0, 0, \ldots, 0) = 0 \tag{25}$$

(2) Rosenbrock function.

$$f_2(x) = \sum_{i=1}^{n-1} \left[100\left(x_{i+1} - x_i^2\right)^2 + (x_i - 1)^2\right], \quad -200 \le x_i \le 200, \quad \min f_2(x) = f_2(1, 1, \ldots, 1) = 0 \tag{26}$$

(3) Griewank function.

$$f_3(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\frac{x_i}{\sqrt{i}} + 1, \quad -600 \le x_i \le 600, \quad \min f_3(x) = f_3(0, 0, \ldots, 0) = 0 \tag{27}$$

(4) Schaffer function.

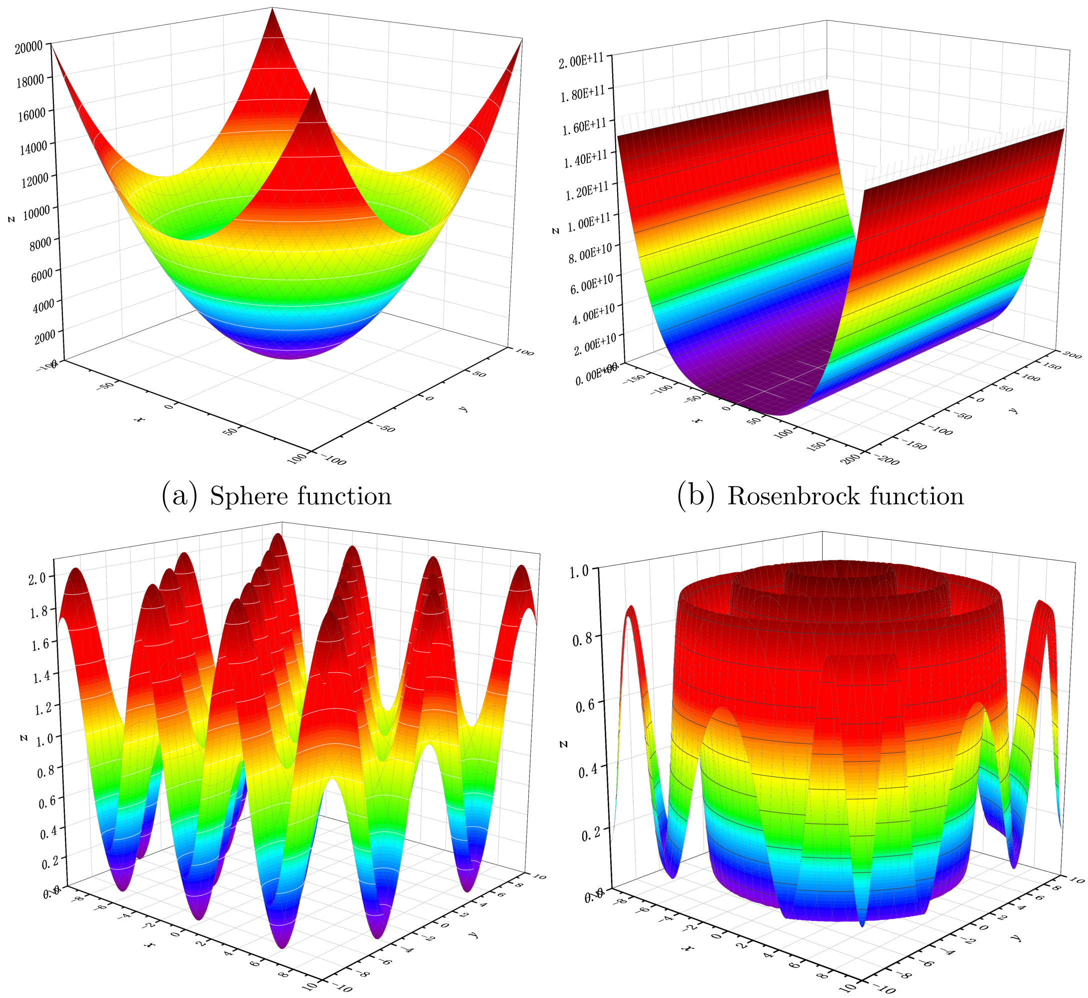
$$f_4(x, y) = 0.5 - \frac{\sin^2\sqrt{x^2 + y^2} - 0.5}{\left(1 + 0.001\left(x^2 + y^2\right)\right)^2}, \quad -10 \le x, y \le 10, \ \min f_4(x, y) = f_4(0, 0) = 1 \tag{28}$$

The geometric curve characteristics of the four functions are shown in Figure 5.

The average optimisation results of the 4 algorithms run 10 times are shown in Table 1.

Table 1. Optimization result

Arithmetic	Indicators	Sphere function	Rosenbrock function	Griewank function	Schaffer function
GA	average value	4.0×10-15	1.3×10-6	1.3×10-7	2.8×10-16
	standard deviation	2.5×10-15	1.5×10-6	3.2×10-7	2.5×10-16
	time/s	0.81	0.15	0.70	0.19
IGA	average value	8.7×10-8	0.1×10-1	3.1×10-8	0.8×10-2
	standard deviation	1.7×10-7	0.9×10-2	3.6×10-8	0.5×10-2
	time/s	10.26	11.20	11.01	0.78
MGA	average value	2.9×10-35	1.9×10-5	2.8×10-28	1.1×10-3
	standard deviation	9.3×10-35	4.8×10-5	3.5×10-28	2.2×10-3
	time/s	12.88	11.52	11.06	0.96
PAGA	average value	0	0	3.1×10-30	0
	standard deviation	0	0	5.5×10-30	0
	time/s	2.16	2.21	2.41	0.46



(a) Sphere function

(b) Rosenbrock function

(c) Griewank function

(d) Schaffer function

Figure 5. Geometric Curve Properties of 4 Functions

As seen from the results of the test function, the PAGA algorithm has the highest overall optimisation accuracy. From the standard deviation, the PAGA algorithm is less volatile. From the running time, the PAGA algorithm is slightly inferior to the standard PSO algorithm, but significantly better than the IGA algorithm and MGA algorithm. Comprehensively analysing the algorithm's optimization accuracy, optimization speed and robustness, the PAGA algorithm is better than the other three algorithms, and thus the PAGA algorithm adapts to a wider range of solving optimization problems, thus providing a strong support for the subsequent performance analysis of the PAGA-HMM model.

5.2. **Data error correction results.** As shown above, the WSN node data is divided into smooth data, and latitude and longitude data. Firstly, the constant temperature T is set to be $3°C$, assuming that the initial temperature T_{k-1} is also $3°C$, and the transport noise T_{oe} is $0.3°C$. Secondly, multiple road segments are selected, and about 200 consecutive positioning points in motion are selected on each road, and the distance in seconds between waypoints is 5s. Ten trajectory datasets are obtained, in which the time is used in the format of Unix timestamps for convenient storage, and some of the data are shown in Table 2.

Table 2. Selected data from the trajectory dataset.

Coordinate system	Latitude	Longitude	Time	Velocity/km-h^{-1}
GCJ02	118.94306578	32.11705188	2022-10-12 08:00	32.21
GCJ02	118.94337778	32.11619665	2022-10-12 08:05	32.17
GCJ02	118.94353583	32.11577970	2022-10-12 08:10	32.83
GCJ02	118.94382966	32.11519097	2022-10-12 08:15	36.16
GCJ02	118.94416870	32.11463515	2022-10-12 08:20	38.63
GCJ02	118.94441540	32.11411700	2022-10-12 08:25	39.51
GCJ02	118.94471602	32.11361997	2022-10-12 08:30	40.07
GCJ02	118.94506685	32.11304509	2022-10-12 08:35	42.23
GCJ02	118.94533914	32.11259178	2022-10-12 08:40	41.65

Firstly, according to the experimental parameters, the smooth data error correction algorithm based on Kalman filtering is used for experiments. The Kalman filtered temperature is calculated based on the initial temperature and the sensor measurement temperature, and the measurement error is analysed, and the results are shown in Figure 6:

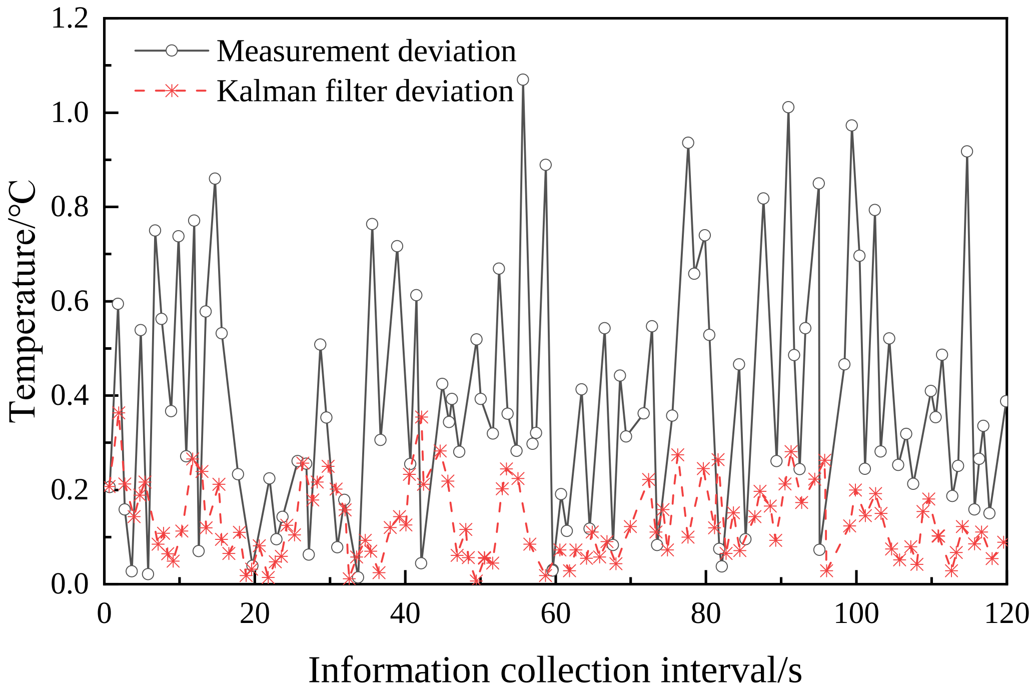


Figure 6. Comparison of temperature error correction deviation

It can be seen that the calculated values using the Kalman filter algorithm are closer to the real values and have less measurement deviation than the data measured directly using the sensors. This indicates that the Kalman filtering algorithm has a good error correction effect on smooth data such as temperature in the cold chain environment.

Then, the effect of introducing polynomial variation operator and adaptive immigration operator on the fitness function of the GA algorithm is analysed, as shown in Figure 7.

It can be seen that under the condition of the same number of iterations, the size of the fitness function of PAGA is higher than that of the other GA and IGA when considering the polynomial variation operator and adaptive immigration operator, which is due to the fact that the excellent individuals are directly retained in the process of population

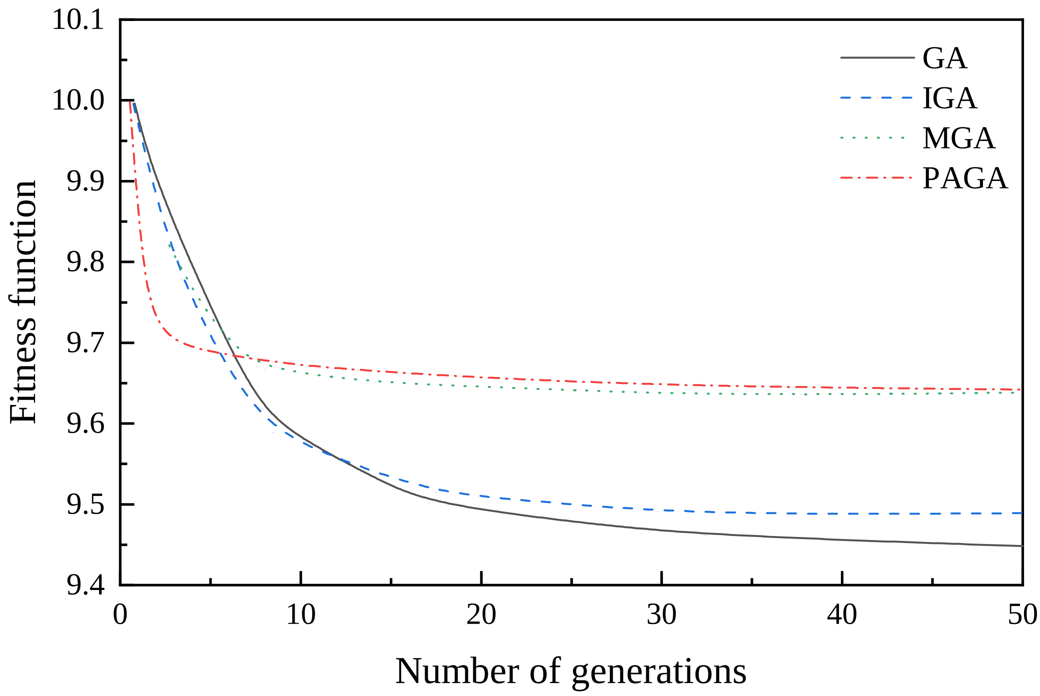


Figure 7. Comparison of the effect of the new operator on the fitness function

evolution. The size of the fitness function is about the same for PAGA and MGA, but the number of iterations is fewer in PAGA, which leads to the smoothness earlier. state. This is because the MGA algorithm only considers polynomial variance and common selection factors, which leads to too much randomness, and although it improves the accuracy of the fitness function, the computation time for a single iteration increases.

A comparison of the convergence of the fitness functions calculated by the Baum-Welch algorithm in the traditional HMM model and the PAGA-HMM model is shown in Figure 8.

It can be seen that the Baum-Welch algorithm in the traditional HMM model computes the value of the fitness function in the interval [-2.29, -1.91], while the PAGA-HMM model computes the value of the fitness function in [-2.12, -1.70]. In contrast, the traditional HMM model is globally unstable and more likely to fall into local optima. However, the PAGA-HMM model has fewer iterations, is faster, and improves the maximum value of the fitness function by 0.21, an improvement of nearly 12%, which is effective and more accurate.

The median filter, HMM model, and PAGA-HMM model are used to calculate the trajectory correction for the positioning data shown in Table 2, respectively. After the experiments conducted on several road sections, the final experimental results are obtained as shown in Figure 9.

It can be seen that the processing of trajectory points with the PAGA-HMM model effectively removes the deviation points, so that the invalid interfering localisation points applied for trajectory matching are removed, thus improving the availability of localisation points. The accuracy of trajectory correction using the PAGA-HMM model is almost always above 90%, and the trajectory accuracy is effectively improved relative to other models, thus verifying the correctness and feasibility of trajectory correction by combining the PAGA algorithm with the HMM model.

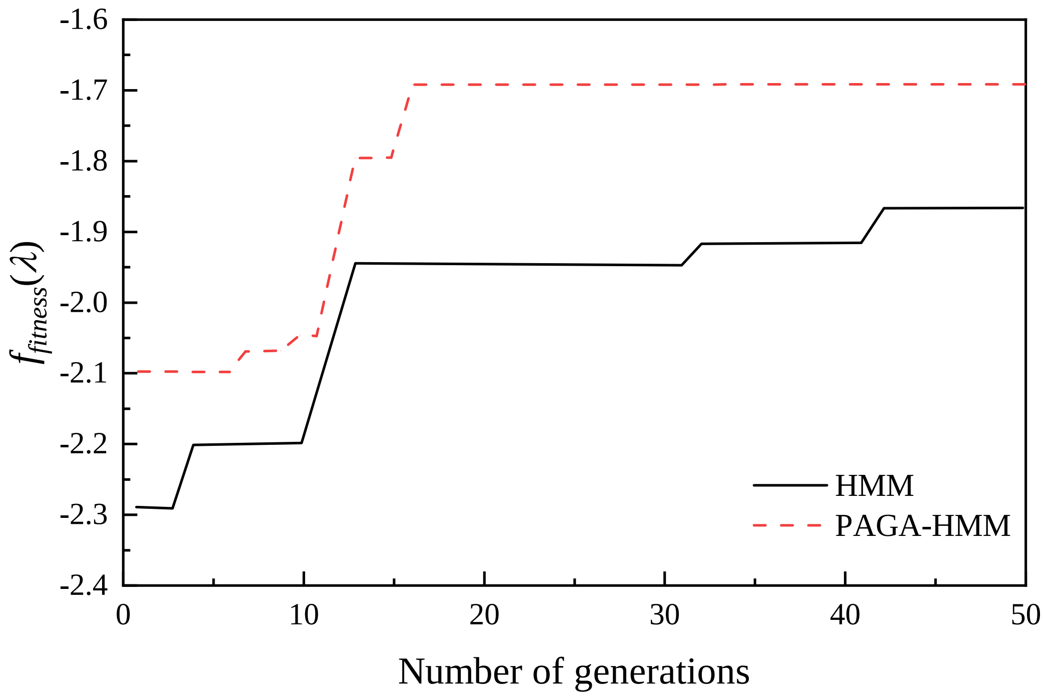


Figure 8. Convergence comparison of fitness function in conventional HMM and PAGA-HMM

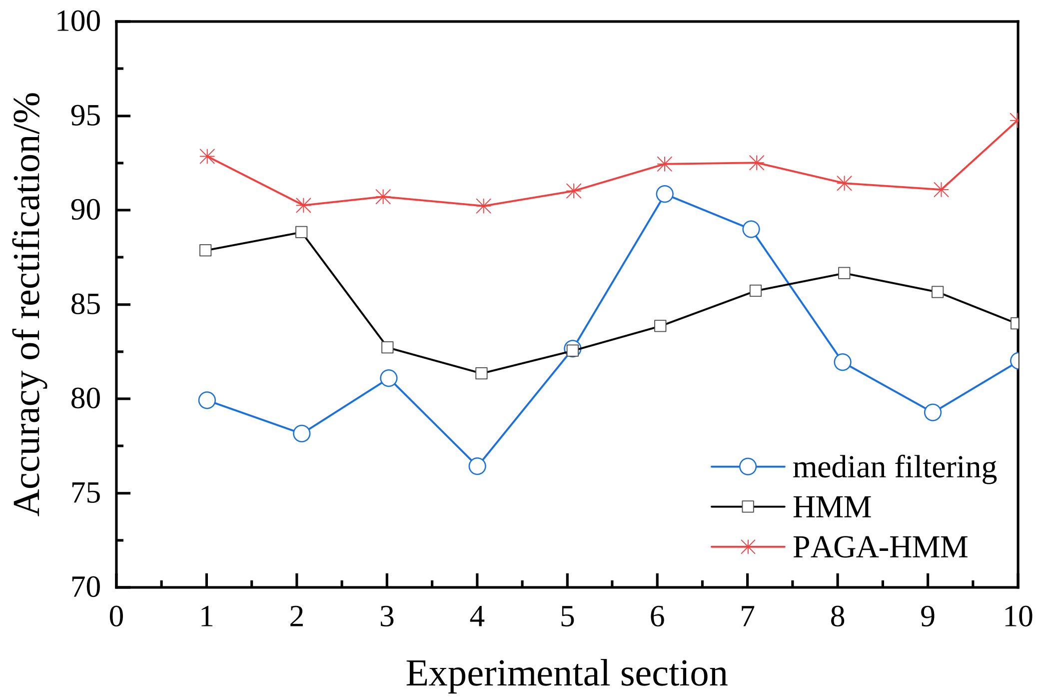


Figure 9. Comparison of correction accuracy

6. **Conclusion.** In the process of cold chain transport, the existence of problems in the sensor output signal and quality may lead to inaccurate data recording, and it is necessary to modify the erroneous data and make an early warning in time. Therefore, for the WSN node data generated during the cold chain transport process, this paper firstly proposes to use the Larman filtering algorithm to correct the errors of these data, which can

make the data curve become smooth and reduce the anomalies, so as to improve the usability of the data. Then, the latitude and longitude data is special, due to its large variation and difficult to speculate, this paper proposes to combine the GA algorithm and HMM model to correct the original trajectory data trajectory. In order to improve the convergence accuracy and speed, a GA algorithm based on polynomial variation and adaptive migration, called PAGA algorithm, is proposed, and its effectiveness is verified by several standard functions. The trajectory correction method based on the PAGA-HMM model is finally constructed. The experimental results show that the accuracy of trajectory correction using the PAGA-HMM model is almost always above 90%. Compared with other models, the trajectory accuracy of the PAGA-HMM model is effectively improved, and it has high reference value for promotion in various application scenarios of IoT.

However, the existing work operates on the assumption that all sensor nodes and devices are working properly. If a sensor node or device fails, the collection and use of data can be greatly affected. Further research on this issue will be conducted in the future.

REFERENCES

[1] T.-Y. Wu, L. Wang, and C.-M. Chen, "Enhancing the Security: A Lightweight Authentication and Key Agreement Protocol for Smart Medical Services in the IoHT," *Mathematics*, vol. 11, no. 17, 3701, 2023.

[2] T.-Y. Wu, Q. Meng, Y.-C. Chen, S. Kumari, and C.-M. Chen, "Toward a Secure Smart-Home IoT Access Control Scheme Based on Home Registration Approach," *Mathematics*, vol. 11, no. 9, pp. 2123, 2023.

[3] T.-Y. Wu, F. Kong, Q. Meng, S. Kumari, and C.-M. Chen, "Rotating behind security: an enhanced authentication protocol for IoT-enabled devices in distributed cloud computing architecture," *EURASIP Journal on Wireless Communications and Networking*, vol. 2023, 36, 2023.

[4] A. Shahraki, A. Taherkordi, O. Haugen, and F. Eliassen, "Clustering objectives in wireless sensor networks: A survey and research direction analysis," *Computer Networks*, vol. 180, 107376,2020.

[5] H. Landaluce, L. Arjona, A. Perallos, F. Falcone, I. Angulo, and F. Muralter, "A Review of IoT Sensing Applications and Challenges Using RFID and Wireless Sensor Networks," *Sensors*, vol. 20, no. 9, 2495, 2020.

[6] X.-N. Li, S.-S. Liu, S. Kumari, and C.-M. Chen, "PSAP-WSN: a provably secure authentication protocol for 5g-based wireless sensor networks," *CMES-Computer Modeling in Engineering & Sciences*, vol. 135, no. 1, pp. 711-732, 2013.

[7] A. Rana, S. Prajapat, P. Kumar, D. Gautam, and C.-M. Chen, "Designing a Security Framework Based on Hybrid Communication in the Internet of Nano Things," *IEEE Internet of Things Journal*, 2023. [Online]. Available: https://doi.org/10.1109/JIOT.2023.3315712

[8] D. K. Sah, and T. Amgoth, "Renewable energy harvesting schemes in wireless sensor networks: A Survey," *Information Fusion*, vol. 63, pp. 223-247, 2020.

[9] H. Mohapatra, and A. K. Rath, "Fault-tolerant mechanism for wireless sensor network," *IET Wireless Sensor Systems*, vol. 10, no. 1, pp. 23-30, 2020.

[10] J. Luo, Y. Chen, M. Wu, and Y. Yang, "A Survey of Routing Protocols for Underwater Wireless Sensor Networks," *IEEE Communications Surveys & Tutorials*, vol. 23, no. 1, pp. 137-160, 2021.

[11] J. Liu, Z. Zhao, J. Ji, and M. Hu, "Research and application of wireless sensor network technology in power transmission and distribution system," *Intelligent and Converged Networks*, vol. 1, no. 2, pp. 199-220, 2020.

[12] U. Saeed, S. U. Jan, Y.-D. Lee, and I. Koo, "Fault diagnosis based on extremely randomized trees in wireless sensor networks," *Reliability Engineering & System Safety*, vol. 205, 107284, 2021.

[13] F. Li, T. Li, H. Wang, and Y. Jiang, "A Temperature Sensor Clustering Method for Thermal Error Modeling of Heavy Milling Machine Tools," *Applied Sciences*, vol. 7, no. 1, 82, 2017.

[14] K. Liu, Y. Liu, M. Sun, X. Li, and Y. Wu, "Spindle axial thermal growth modeling and compensation on CNC turning machines," *The International Journal of Advanced Manufacturing Technology*, vol. 87, no. 5-8, pp. 2285-2292, 2016.

[15] H. Kim, Y. Min, and B. Choi, "Real-time temperature monitoring for the early detection of mastitis in dairy cattle: Methods and case researches," *Computers and Electronics in Agriculture*, vol. 162, pp. 119-125,2019.

[16] Y. Lv, Y. Liu, and J. Hua, “A Study on the Application of WSN Positioning Technology to Unattended Areas,” *IEEE Access*, vol. 7, pp. 38085-38099, 2019.

[17] A. Alabadleh, S. Aljaafreh, A. Aljaafreh, and K. Alawasa, “A RSS-based localization method using HMM-based error correction,” *Journal of Location Based Services*, vol. 12, no. 3-4, pp. 273-285, 2018.

[18] H. Kalkha, H. Satori, and K. Satori, “Preventing Black Hole Attack in Wireless Sensor Network Using HMM,” *Procedia Computer Science*, vol. 148, pp. 552-561, 2019.

[19] H. Xiong, X. Huang, M. Yang, L. Wang, and S. Yu, “Unbounded and Efficient Revocable Attribute-Based Encryption with Adaptive Security for Cloud-Assisted Internet of Things,” *IEEE Internet of Things Journal*, vol. 9, no. 4, pp. 3097-3111, 2022.

[20] H. Xiong, Y. Bao, X. Nie, and Y. I. Asoor, “Server-Aided Attribute-Based Signature Supporting Expressive Access Structures for Industrial Internet of Things,” *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1013-1023, 2020.

[21] H. Xiong, Q. Mei, Y. Zhao, L. Peng, and H. Zhang, “Scalable and Forward Secure Network Attestation with Privacy-Preserving in Cloud-Assisted Internet of Things,” *IEEE Sensors Journal*, vol. 19, no. 18, pp. 8317-8331, 2019.

[22] Q. Deng, and D. Soffker, “A Review of HMM-Based Approaches of Driving Behaviors Recognition and Prediction,” *IEEE Transactions on Intelligent Vehicles*, vol. 7, no. 1, pp. 21-31, 2022.

[23] M. Hamidi, O. Zealouk, H. Satori, N. Laaidi, and A. Salek, “COVID-19 assessment using HMM cough recognition system,” *International Journal of Information Technology*, vol. 15, no. 1, pp. 193-201, 2022.

[24] P. Kumar, H. Gauba, P. P. Roy, and D. P. Dogra, “Coupled HMM-based multi-sensor data fusion for sign language recognition,” *Pattern Recognition Letters*, vol. 86, pp. 1-8, 2017.

[25] C. K. H. Lee, “A review of applications of genetic algorithms in operations management,” *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 1-12, 2018.

[26] D. Corus, D.-C. Dang, A. V. Eremeev, and P. K. Lehre, “Level-Based Analysis of Genetic Algorithms and Other Search Processes,” *IEEE Transactions on Evolutionary Computation*, vol. 22, no. 5, pp. 707-719, 2018.

[27] C. Puspitasari, R. Nur, and Wahyono, “Prediction of ozone (o3) values using support vector regression method,” *Jurnal Informatika Polinema*, vol. 7, no. 4, pp. 81-88,2021.

[28] Y. Song, F. Wang, and X. Chen, “An improved genetic algorithm for numerical function optimization,” *Applied Intelligence*, vol. 49, no. 5, pp. 1880-1902, 2018.

[29] W. E. Peterman, “ResistanceGA: An R package for the optimization of resistance surfaces using genetic algorithms,” *Methods in Ecology and Evolution*, vol. 9, no. 6, pp. 1638-1647, 2018.

[30] S. Katoch, S. S. Chauhan, and V. Kumar, “A review on genetic algorithm: past, present, and future,” *Multimedia Tools and Applications*, vol. 80, pp. 8091-8126, 2021.

[31] B. Zhao, H. Gui, H. Li, and J. Xue, “Cold Chain Logistics Path Optimization via Improved Multi-Objective Ant Colony Algorithm,” *IEEE Access*, vol. 8, pp. 142977-142995, 2020.

[32] D. R. K. Srivastava, and D. Pandey, “Speech recognition using HMM and Soft Computing,” *Materials Today: Proceedings*, vol. 51, pp. 1878-1883, 2022.

[33] M. Kaveh, M. Kaveh, M. S. Mesgari, and R. S. Paland, “Multiple criteria decision-making for hospital location-allocation based on improved genetic algorithm,” *Applied Geomatics*, vol. 12, no. 3, pp. 291-306, 2020.

[34] Q. Liu, X. Li, L. Gao, and Y. Li, “A Modified Genetic Algorithm with New Encoding and Decoding Methods for Integrated Process Planning and Scheduling Problem,” *IEEE Transactions on Cybernetics*, vol. 51, no. 9, pp. 4429-4438, 2021.