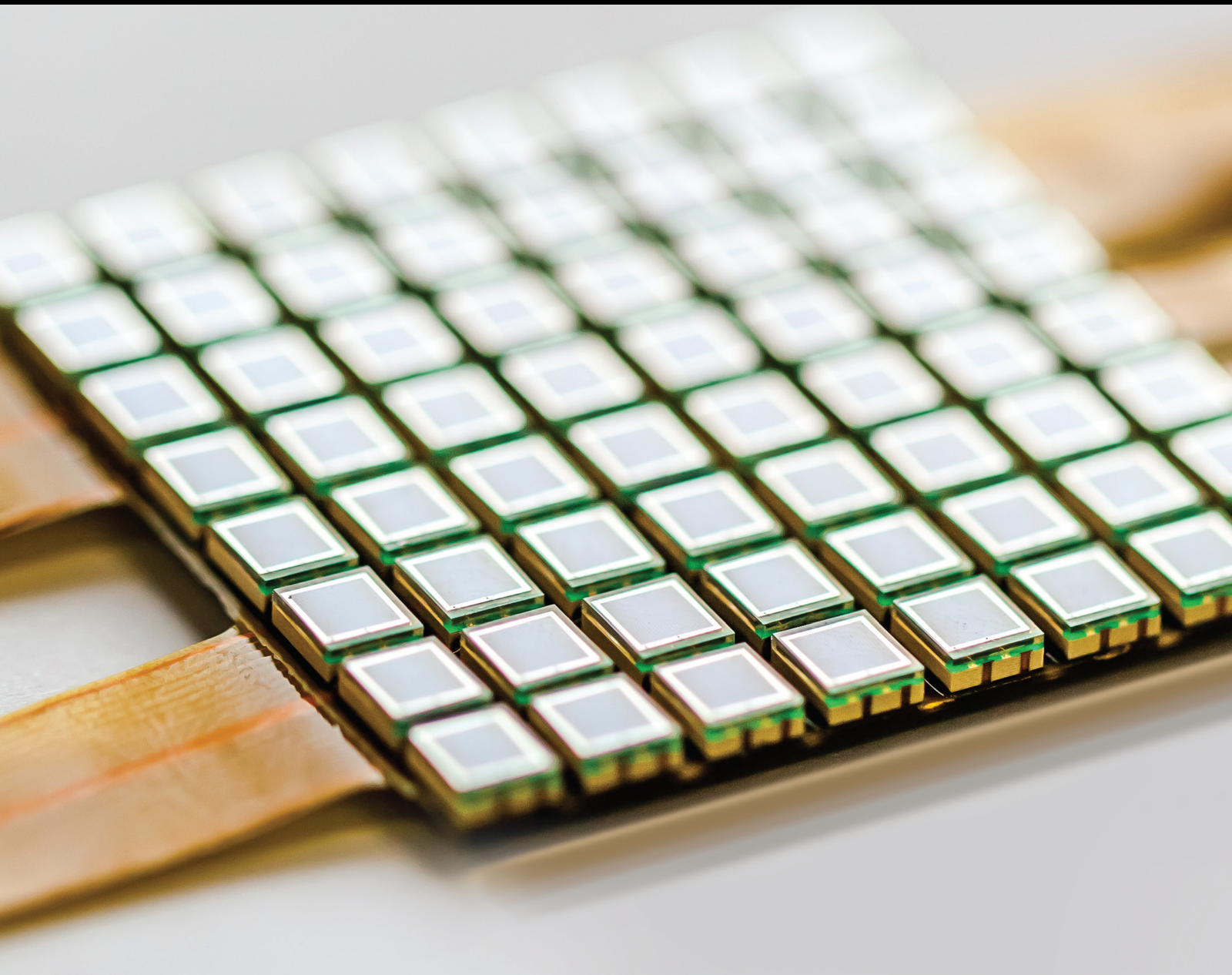# Estimation, Location, Tracking and Control in Complex Multi-Sensor Systems

Lead Guest Editor: Xue-bo Jin
Guest Editors: Jian Lan, Chun-Xi Yang, Quanmin Zhu, and Bingxian Mu

Estimation, Location, Tracking and Control in Complex Multi-Sensor Systems

Estimation, Location, Tracking and Control in Complex Multi-Sensor Systems

Lead Guest Editor: Xue-bo Jin
Guest Editors: Jian Lan, Chun-Xi Yang, Quanmin Zhu, and Bingxian Mu

Contents

Research Article

Jammer Location-Aware Method in Wireless Sensor Networks Based on Fibonacci Branch Search

Fang Yang ⓘ, Nina Shu ⓘ, Chenxi Hu ⓘ, Jun Huang, and Zhao Niu

National University of Defense Technology, College of Electronic Engineering, Hefei, Anhui 230037, China

Correspondence should be addressed to Nina Shu; shunina@nudt.edu.cn

Due to the sharing and open-access characteristics of the wireless medium, wireless sensor networks (WSNs) can be easily attacked by jammers. To mitigate the effects of a jamming attack, one reliable solution is to locate and remove the jammer from the deployed area within the WSN. To realize the jammer's localization in the WSN, many range-free methods have been proposed. However, most of these methods are sensitive to the distribution of nodes and the parameters of the jammer. For this reason, a jammer location-aware method based on Fibonacci branch search (FBS) is proposed in this article. First, the interference region is estimated by using the interference region mapping service of sensors in wireless sensor networks. Then, the search point is selected in the jamming area and the fitness function is designed according to the average distance from the search point to the boundary sensor. According to the basic branch structure and interactive search rules, the global optimal solution is obtained in the jamming area. Finally, the position of the search point with the best fitness value is used as the estimation of the jammer position. Compared with the existing typical range-free methods, rich simulation experiments demonstrate that the FBS algorithm is superior in the location-aware method for jammers with a higher precision and a lower sensitivity to the distribution of nodes and the parameters of the jammer, respectively.

1. Introduction

Wireless sensor networks (WSNs) are seriously threatened by radio interference attacks due to the sharing and open-access characteristics of wireless mediums [1–3]. The radio interference attacks, which are also known as jamming attacks, can seriously disrupt normal communication between legitimate sensors. By occupying the wireless communication channel or disrupting the workflow of network protocols, WSN jamming attacks can be easily initiated [4]. To reduce the impact of jamming attacks on network performance and ensure the security of WSNs, various prevention jamming strategies [5] have been proposed, such as covert timing channels [6], channel hopping [7], protocol optimization [8], channel-aware decision fusion [9], and spatial retreat [10]. In addition to these strategies, jammer location awareness is an effective method that helps us remove the jammer based on the jammer's location [11].

To date, most jammer location-aware problems have been extensively investigated and several location perception

algorithms have been designed. In general, existing location-aware strategies can be classified as range-free methods and range-based methods [12]. Range-based methods usually estimate the distance information by measuring some physical attributes of jammer signals [13]. The relevant physical attributes mainly include received signal strength indicators (RSSIs) [14], time difference of arrival (TDoA) [15], time of arrival (ToA) [16], and angle of arrival (AoA) [17]. The location of the jammer is calculated by the Euclidean distance or angle size between legitimate nodes and the jammer. Different calculation algorithms will greatly affect the positioning accuracy. Typical algorithms include triangulation [18], trilateration [19], and algorithms based on multidimensional scaling (MDS), such as MDS [20], MDS-MAP [21], MDS-MAP(P) [22], and SMDS [23]. However, due to the small size of sensors in WSNs, battery life is limited and the life of the sensor is directly affected by the life of the battery [24], which further affects the life of the entire network to maximize the life cycle of the sensor network. Therefore, the seniors in WSNs are usually not allowed components

that can realize the distance between the sensors and the jammer. At the same time, sensors in the jammed area have difficulty communicating with each other because of jammer interference. Even though every sensor is equipped with ranging components, the jammed sensor cannot obtain the location information of neighboring sensors to locate the jammer's position. For these reasons, range-free methods are more suitable for WSNs to achieve the localization of jammers in the network [25].

Range-free location-aware algorithms make use of the geometric relationship of the jammed area to realize the localization of the jammer. At the same time, the problem of jamming detection was briefly researched by Wood et al. [26] and further investigated by Xu et al. [27]. Wood et al. proposed a scheme that could map a jammed region, and Xu et al. presented several measurements to detect jamming attacks in WSNs. To calculate the jammer localization, Bulusu et al. proposed centroid localization (CL) [28], which estimates the coordinates of the jammer by calculating the average relative coordinates of all jammed nodes. CL is easy to realize, but the localization error is large. To improve the location-aware accuracy, Blumenthal et al. proposed weighted centroid localization (WCL) [29], which is based on the assumption that the influence degree of different nodes on the localization of the jammer is different. The closer the node is to the jammer, the greater the weight value of the node is. WCL needs RSSI to realize the computation of the weight, which is difficult in some scenarios. To further improve the location-aware accuracy, Shoari and Seyedi proposed an algorithm based on the minimum enclosing rectangle center [12]. In the algorithm, the position of the jammer is calculated as the center of the smallest enclosing rectangle covering all of the jamming nodes. Liu et al. proposed virtual force iterative localization (VFIL) to locate the jammer [30]. The concepts of pull force, push force, and joint force are defined in the VFIL. The jammed node produces a pulling force to constantly pull the jammer to itself, and the boundary sensors push the jammer away from itself through a push force. The final location awareness of the jammer is realized through the joint force produced by the jammed nodes and the boundary nodes in the jammed area. To reduce the sensitivity of the distribution of nodes when locating the jammer, Wang et al. proposed a heuristic optimization evolutionary algorithm named the gravitational search algorithm (GSA) [1]. This method merges mass interactions and Newton's law of universal gravitation. After iterations, the coordinates of the most massive particle are adopted as the coordinates of the jammer. In addition, for multijammer scenarios, Cheng et al. designed two methods to calculate the localization of jammers based on M-clusters and X-rays [31]. Wang et al. utilized the k-means clustering algorithm to estimate the location of the jammer according to the location information of neighboring nodes [32].

The distributed sensors and the parameters of the jammer can easily affect the performance of range-free methods. In this paper, to decrease the sensitivity of range-free methods and improve the location-aware accuracy, a robust location-aware algorithm based on the Fibonacci branch search (FBS) is designed. Meanwhile, the reachability and convergence of FBS are proven mathematically, which further verifies the validity of the theory for the FBS strategy. The location-aware algorithm based on FBS uses the powerful global searchability and the high convergence speed of the technology. It improves the location-aware performance by preventing the loss of the best trajectory. Aiming at the abovementioned problems in target node location, the main contribution of this paper includes three aspects.

(1) In this paper, a robust location sensing algorithm based on Fibonacci branch search (FBS) is designed to reduce the sensitivity of distance-independent methods and improve the accuracy of location sensing. At the same time, the reachability and convergence of FBS are proved mathematically, which further verifies the effectiveness of the FBS strategy theory

(2) Based on the FBS algorithm, interactive global search and local optimization rules are used alternately to realize global optimization. Finally, the coordinates of the search point with the highest fitness value are taken as the coordinates of the jammer. Compared with many existing location aware algorithms, the proposed method has higher performance in complex scenes with different parameter settings

(3) Experiments show that the proposed method can locate the target node when it is unable to range the target node, and in terms of optimization ability, due to other similar algorithms, at the same time, even if the deployment area of the wireless sensor network has the characteristics of low density and low communication distance, the positioning error based on FBS is still less than other algorithms

The rest of this paper is described as follows. Section 2 describes the network model and the jamming model. Section 3 illustrates the main principle of the FBS algorithm. The location-aware method of jammers based on FBS is presented in Section 4. Section 5 illustrates and discusses the simulation results. Section 6 presents our conclusions.

2. System Models

The scenario with a WSN and a jammer is shown in Figure 1, where the jammer is surrounded by sensors. In this section, the network model and the jamming model are outlined.

2.1. Network Model. Assume that N_S homogeneous sensors are deployed in the area to form a WSN. $\mathbf{S} = [\mathbf{s}_1, \mathbf{s}_2, \cdots, \mathbf{s}_{N_S}]^T \in N_S \times 2$ are the coordinates of all of the sensors, and $\mathbf{s}_i = [x_i^s, y_i^s], i = 1, 2, \cdots, N_S$. The sensors can be aware of their locations through GPS or other location-aware algorithms, e.g., when the distances between the sensors are obtained, algorithms based on MDS can be used to realize location awareness. Once deployed, this article considers the location of the sensor to remain unchanged. Every sensor in the network is equipped with an omnidirectional
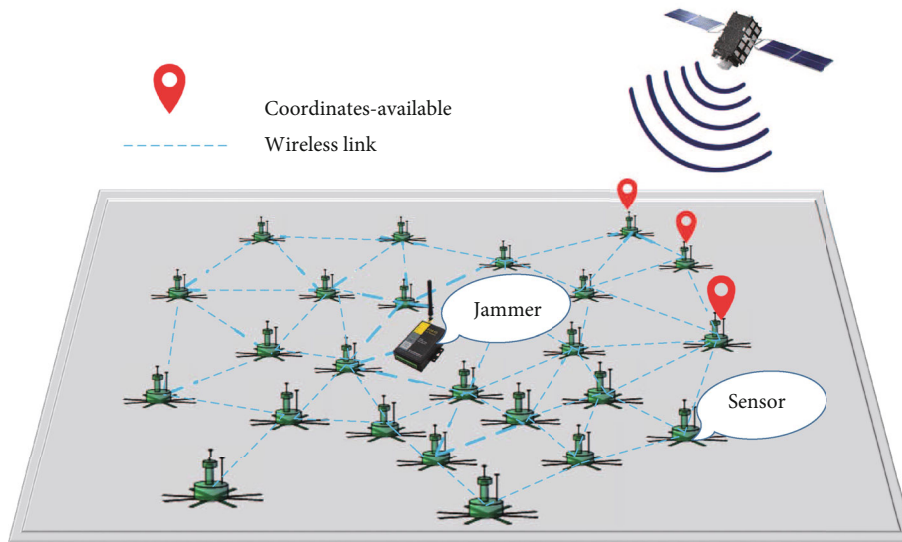
FIGURE 1: Scenario diagram of a WSN and a jammer.

antenna and can communicate with other sensors within a communication range R_S. There are no isolated subnets or isolated sensors in the network. For sensors that are not in the communication range, sensors can communicate with each other in a multihop manner. According to the abovementioned assumptions, the network model of the WSN in Figure 1 is shown in Figure 2.

2.2. Jamming Model. In this model, we assume that the jamming power remains stable and the jammer location stays static. This jammer continuously transmits radio signals, which can be implemented using a waveform generator that continuously transmits radio signals or a normal wireless device that keeps sending random bits to the channel without following any MAC layer protocol. Due to the large coverage of the omnidirectional antenna, the jammer is equipped with an omnidirectional antenna with the same direction effect. Under the effect of the jammer, the sensors in the WSN are divided into three types: jammed sensors, boundary sensors, and unaffected sensors. Jammed sensors are located in the jammed area and cannot communicate with any neighboring sensors. Boundary sensors are usually located at the edge of the jammed area. Although they struggle with jamming attacks, the sensors can still communicate with neighboring sensors. Unaffected sensors outside the jammed area can receive information from neighboring sensors even if the jammer appears. $\mathbf{B} = [\mathbf{b}_1, \mathbf{b}_2, \cdots, \mathbf{b}_{N_B}]^T \in N_B \times 2$ contains the coordinates of all of the boundary sensors, and $\mathbf{b}_i = [x_i^B, y_i^B]$, $i = 1, 2, \cdots, N_B$, where N_B is the number of boundary sensors in the network.

According to the abovementioned assumptions, the jamming model of the WSN is shown in Figure 3.

When a node detects itself as jammed, the node broadcasts notification messages to its neighboring nodes, as shown in Figure 4(a). Mapping is conducted by the neighboring sensors of jammed sensors who receive the interfer-
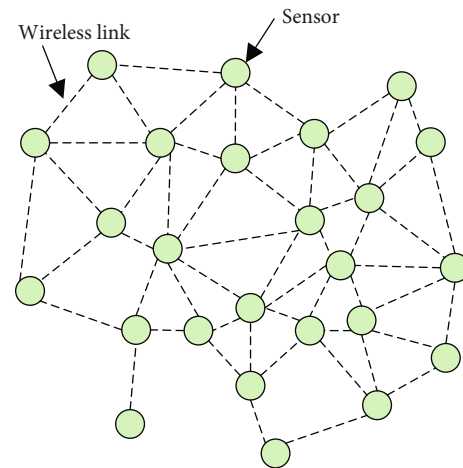
FIGURE 2: The network model of the WSN.

ence message. Each receiver becomes a mapping member and adds nearby jammed sensors to form a local group. As shown in Figure 4(b), adjacent sensors contain mapping messages for the local exchange of group information. Adjacent groups are condensed together to form a mapped area, as shown in Figure 4(c), which can be used as an estimation of the jammed area [26]. The sensors that constitute the mapped area are all boundary nodes. The notations that will be used throughout the paper are summarized in Table 1.

3. Fibonacci Branch Search Algorithm

The Fibonacci optimization method has proven the effectiveness and convergence of solving a series of nonlinear benchmark functions in one-dimensional space [33]. However, the method is rarely used in the properties of multidimensional space search optimization problems. In addition to the structure itself, there are few variants implemented in localized applications reported in the public literature.
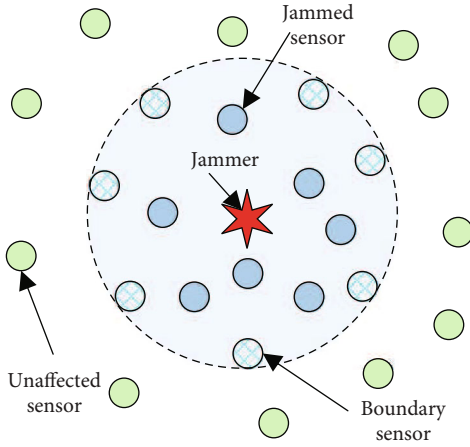
FIGURE 3: The jamming model of the WSN.

In this section, we will briefly introduce the traditional Fibonacci sequence method and explain the principle and in-depth understanding of the FBS-based method.

3.1. Basic Principle of the Fibonacci Method. The Fibonacci sequence, also known as the golden section sequence, was first proposed by Fibonacci [34], and its general formula is as follows [35].

$$
\begin{aligned}
F(1) = F(2) &= 1, \\
F(j) = F(j-1) + F(j-2), \quad j \geq 3,
\end{aligned}
\tag{1}
$$

where F_j indicates the jth Fibonacci sequence, which is a general term.

This paper studies the optimization method of the Fibonacci sequence to solve the minimization problem of a unimodal continuous function in the interval. The Fibonacci sequence optimization method compresses the search interval proportionally according to the Fibonacci sequence items. Then, the initial optimization point converges to the best method in a defined interval, which is considered to be the most effective solution of a one-dimensional unimodal question. We assume that there is a single-peak function on the interval. First, the technique starts to select two feasible points and performs the first iteration within a given range. Then, we need to reduce the area of the initial box to a sufficiently small box, involving the minimum value of the unimodal function $f(x)$ (after an iterative step). The smallest lie can be reduced by providing the function value known in two different ranges of points. The realization of the classic Fibonacci sequence optimization algorithm was given in [36], and no further detailed description is given here.

3.2. The Fibonacci Branch Structure and the FBS Algorithm. The basic Fibonacci strategy has difficulty effectively solving the multivariable problem, nor can it reliably evaluate the best fit of the multimodal function [35]. The FBS algorithm mainly uses the framework constructed by segmentation points and endpoints in the basic structure to search for the global optimal solution. Comparing the fitness value of

each search point by calculation, the search point closest to the global optimal solution is obtained. In the next iterative calculation, we set the point with the best fitness value obtained in this optimization at the top of the search point set and the points corresponding to the suboptimal fitness value are arranged in the order from good to bad below the best. Through continuous iteration, the search point set is updated in each optimization stage. Then, the algorithm can complete the optimization of the objective function in the search space while the Fibonacci branch grows. The basic structure of the Fibonacci algorithm is shown in Figure 5.

In Figure 5, there are three points in D-dimensional Euclidean space, $\vec{x}_A$, $\vec{x}_B$, and $\vec{x}_C$. $\vec{x}_A$ and $\vec{x}_B$ indicate the coordinates corresponding to the search endpoint of the tree structure, which can be generated by the specified optimization rule, and $\vec{x}_C$ indicates the coordinates of the partition point obtained according to the given calculation criteria. In the search process, they should satisfy the following equation:

$$
\frac{\left\| \vec{x}_C - \vec{x}_A \right\|}{\left\| \vec{x}_B - \vec{x}_A \right\|} = \frac{\left\| \vec{x}_B - \vec{x}_C \right\|}{\left\| \vec{x}_C - \vec{x}_A \right\|} = \frac{F_p}{F_{p+1}},
\tag{2}
$$

where F_p represents the pth Fibonacci number.

Considering the minimum multimodal function of multivariable $f(\vec{X})$ in the search space, the calculation formula of the split point is

$$
\vec{x}_C = \begin{cases} \vec{x}_A + \dfrac{F_p}{F_{p+1}} \left(\vec{x}_B - \vec{x}_A \right), & f\left(\vec{x}_A \right) < f\left(\vec{x}_B \right), \\[3mm] \vec{x}_A + \dfrac{F_p}{F_{p+1}} \left(\vec{x}_A - \vec{x}_B \right), & f\left(\vec{x}_A \right) \geq f\left(\vec{x}_B \right). \end{cases}
\tag{3}
$$

In [37], a similar algorithm was described but the process and theory of the algorithm were not described in detail. This section expounds on the main part of FBS, expounds on the implementation content of FBS, and standardizes the implementation process of FBS.

Considering the basic structure of FBS, the process of obtaining the global optimal solution, including the process of building searching elements in FBS, is delimited into two stages: global search and local optimization [38]. These two stages are the corresponding rules of interaction. $\mathbf{G}_p$ is the set of objective function points to be searched in the pth iterative optimization stage; $\text{len}(\mathbf{G}_p) = F_p$ is the number of whole sets, where F_p is the intensity of the Fibonacci wave. Using the corresponding interactive optimization initial value and fitness value, the segmentation points are obtained through formula (3). After a comparison, the algorithm obtains the best fitness value corresponding to the latest best solution. In the next iteration stage, the best advantage of the adaptable value is concentrated in the corresponding front of the set and the nodes of the suboptimal adaptable value are

(a) Broadcast notification message

(b) Transmission of mapping messages

(c) Construction of a mapped area
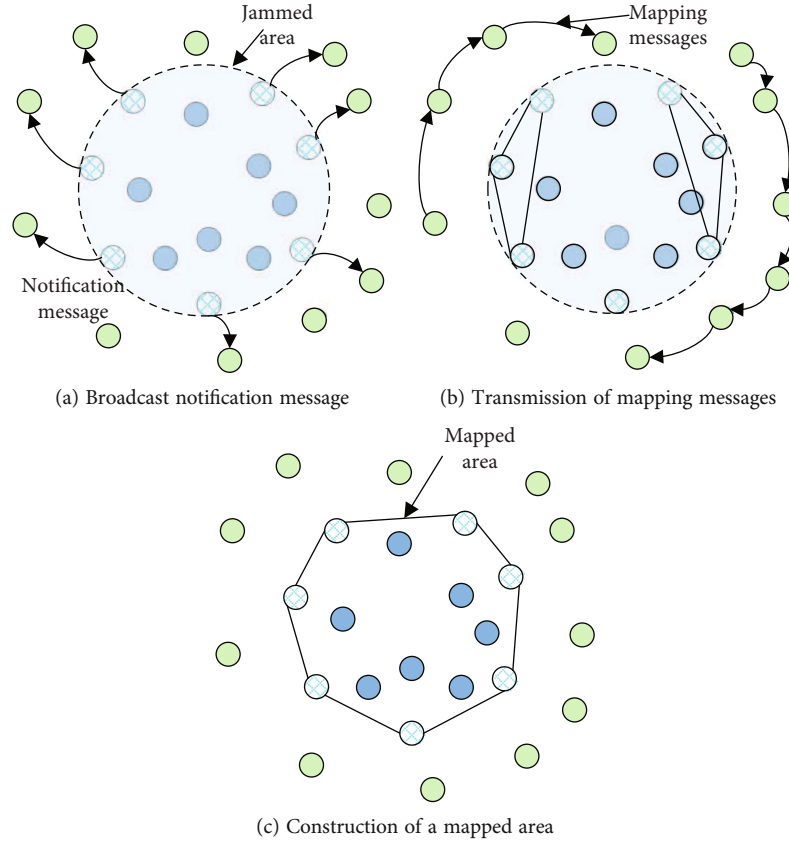
Figure 4: Overview of sensors collaboratively mapping a jammed area.

Table 1: Notations and definitions.

Notations	Definitions
N_S	The number of homogeneous sensors
$\mathbf{S}$	The coordinates of all of the sensors
$\mathbf{B}$	The coordinates of all of the boundary sensors
N_B	The number of boundary sensors
R	The branch depth
F_j	The jth Fibonacci sequences
$\mathbf{G}_p$	The set of objective function points to be searched in the pth iterative optimization stage
$\hat{\mathbf{r}}$	The estimated result of the jammer's coordinates
$\mathbf{r}$	The real location of the jammer
$\vec{x}_A, \vec{x}_B$	Search endpoint
$\vec{x}_C$	Split point

placed in the order from good to bad in the optimal selection. Through the abovementioned operations, we update points in each optimization stage and increase the Fibonacci path and community optimization search space.

In the process of FBS optimization, it is necessary to update the search endpoint according to two interactive iterative criteria and combine the calculation formula of the subdivision point to calculate the subdivision point. The two iterative updating criteria are as follows:

In rule one, the end nodes $\vec{x}_A$ and $\vec{x}_B$ of the structure are indicated as follows:

$$\left\{\vec{x}_A\right\} = G_p = \left\{\vec{x}_q | q = [1, F_P]\right\},$$
$$\left\{\vec{x}_B\right\} = \left\{\vec{X} \middle| \vec{X} \in \prod_{f=1}^{D} \left[\vec{x}_{lb}^{f}, \vec{x}_{ub}^{f}\right]^u\right\}, \tag{4}$$

$\mathbf{G}_p$ is composed of the coordinates of all the search points in the pth iteration, $\vec{x}_q$ are the search nodes in set $\mathbf{G}_p$, and q is the sequence number corresponding to the first to the pth Fibonacci sequences. $\vec{x}_A$ takes all points of $\mathbf{G}_p$ in the pth iteration. The other unselected endpoints $\vec{x}_B$ randomly take nodes, and the length of $\vec{x}_B$ is equivalent to F_p. When the dimension is f, D is the dimension of the points and the search nodes are between $\vec{x}_{lb}^{f}$ and $\vec{x}_{ub}^{f}$. Given that $\forall \vec{x} \in \{\vec{x}_B\}$, the component x in point $\vec{X}$ is a random variable uniformly distributed in the interval $[\vec{x}_{lb}, \vec{x}_{ub}]^U$, in which the normal characteristic U represents the uniform distribution of the variable, and the probability distribution of the component can be calculated as
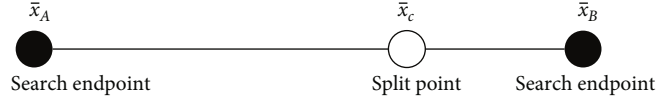
FIGURE 5: Basic search structure of the proposed Fibonacci branch search.
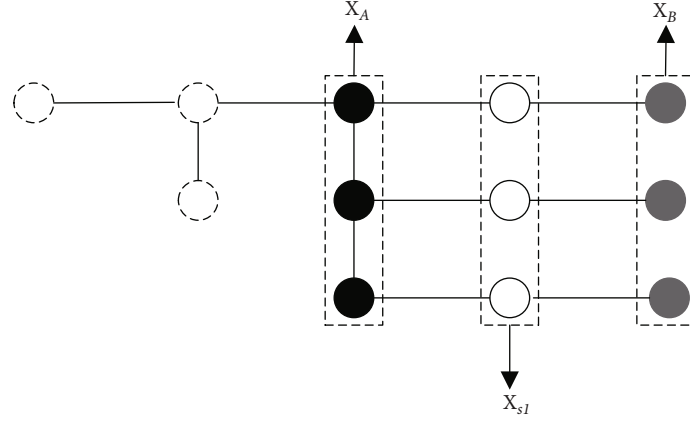
(a) Fibonacci scatter search global search phase
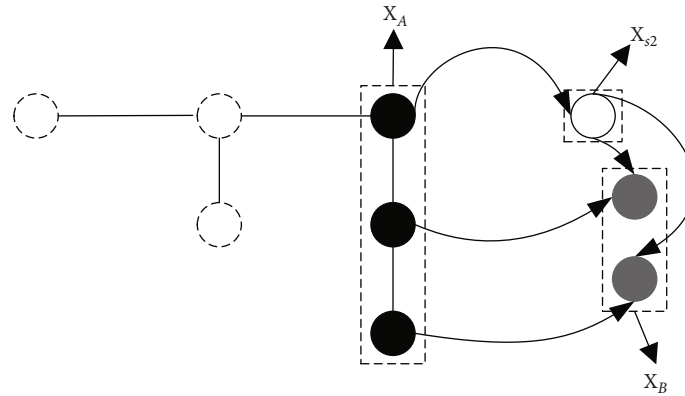
(b) Local optimization phase of Fibonacci scatter search

FIGURE 6: Schematic diagram of the Fibonacci rambler tectonic process.

$$P(x) = U\left(\overrightarrow{x}_{lb}, \overrightarrow{x}_{ub}\right) = \frac{1}{\overrightarrow{x}_{ub} - \overrightarrow{x}_{lb}}. \tag{5}$$

We can use the end nodes $\overrightarrow{x}_A$ and $\overrightarrow{x}_B$ to determine the partition points $\overrightarrow{x}_{S1}$ through equation (3).

In rule two, in the local optimization stage, assume that $\overrightarrow{x}_{best}$ represents the search point with the optimal fitness value in the iterative process of the algorithm, as follows:

$$\overrightarrow{x}_{best} = best\left(\mathbf{G}_p\right). \tag{6}$$

Best($\cdot$) is the optimal fitness value search node in the search node set.

Afterwards, we define the end nodes $\overrightarrow{x}_A = \overrightarrow{x}_{best}$ and find

$$f\left(\overrightarrow{x}_A\right) = \min\left\{f\left(\overrightarrow{x}_q\right), q = \left[1, F_p\right]\right\},$$
$$\overrightarrow{x}_B = \left\{\overrightarrow{x}_q \middle| \overrightarrow{x}_q \in \mathbf{G}_p \wedge \overrightarrow{x}_q \neq \overrightarrow{x}_A\right\}. \tag{7}$$

According to the end points defined in formula (7), the partition point $\overrightarrow{x}_{S2}$ can be determined according to the split point calculation formula in the second local optimization stage.

According to the abovementioned two interactive search rules, two different optimization stages generate $3F_p$ new points, involving endpoints $\overrightarrow{x}_A$ and $\overrightarrow{x}_B$ and segmentation points $\overrightarrow{x}_{S1}$ and $\overrightarrow{x}_{S2}$. By assessing the cost function of the new nodes, the fitness of the new points is determined and the new points are ranked from good to bad according to
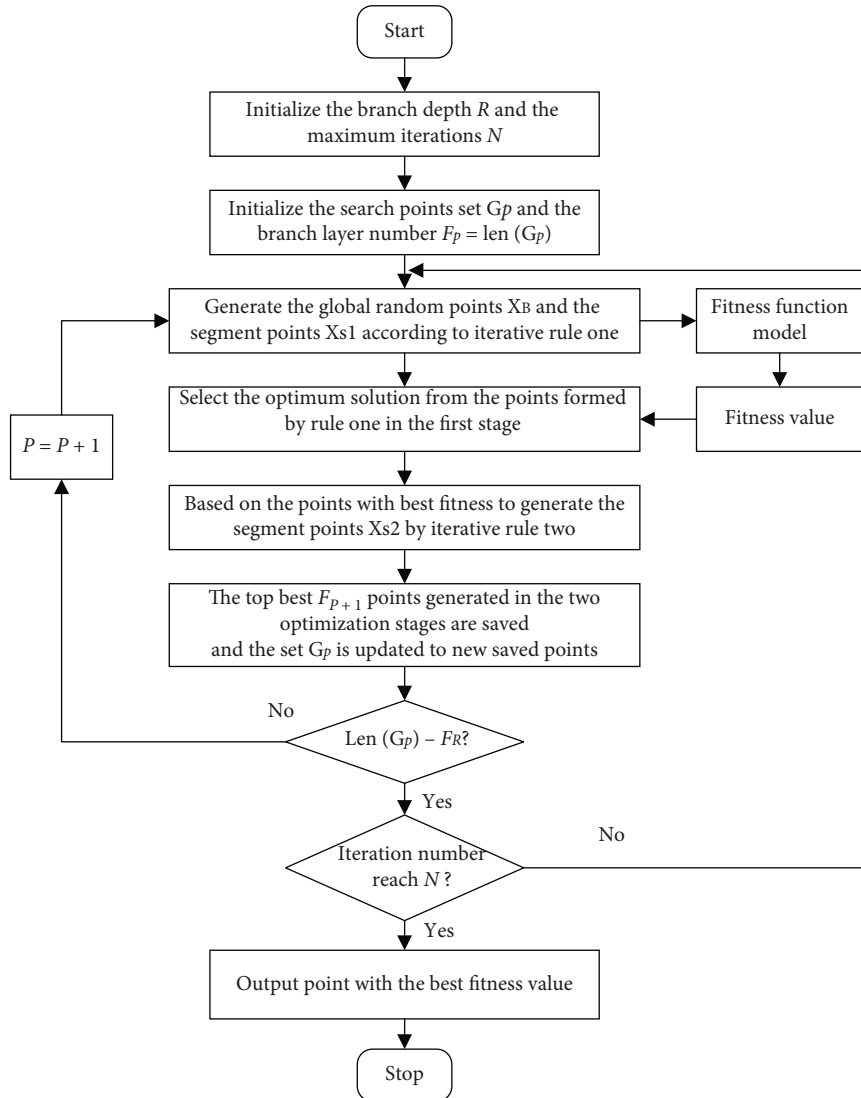
FIGURE 7: Flowchart of the Fibonacci branch search algorithm.

Input: B, R, N,$\mathbf{G}_p$
Output: $\hat{\mathbf{r}}$
$F_p \longleftarrow \text{len}(\mathbf{G}_p)$
for $i = 1 : N$ do
  for $j = p : R$ do
      Create the overall random nodes $\vec{x}_B$ and the split points $\vec{x}_{S1}$ by the rule one
      Calculate the fitness value for each search point based on the coordinates of the search point to all sensors in **B**
      Find the best result from the nodes formed by rule one
      Based on the points with optimum fitness value to adaptable points $\vec{x}_{S2}$ by rule two
    Sort the all-search points from good to bad according to the fitness value of every search point
    The top best F_{j+1} points are retained and the search points set $\mathbf{G}_j$ is renewed to the new retained points
  end for
  end for

ALGORITHM 1: Location-aware of the jammer based on FBS in WSNs.

(a) 50 iterations

(b) 100 iterations

(c) 200 iterations

(d) 400 iterations

(e) 50 iterations

(f) 100 iterations

Figure 8: Continued.

(g) 200 iterations

(h) 400 iterations
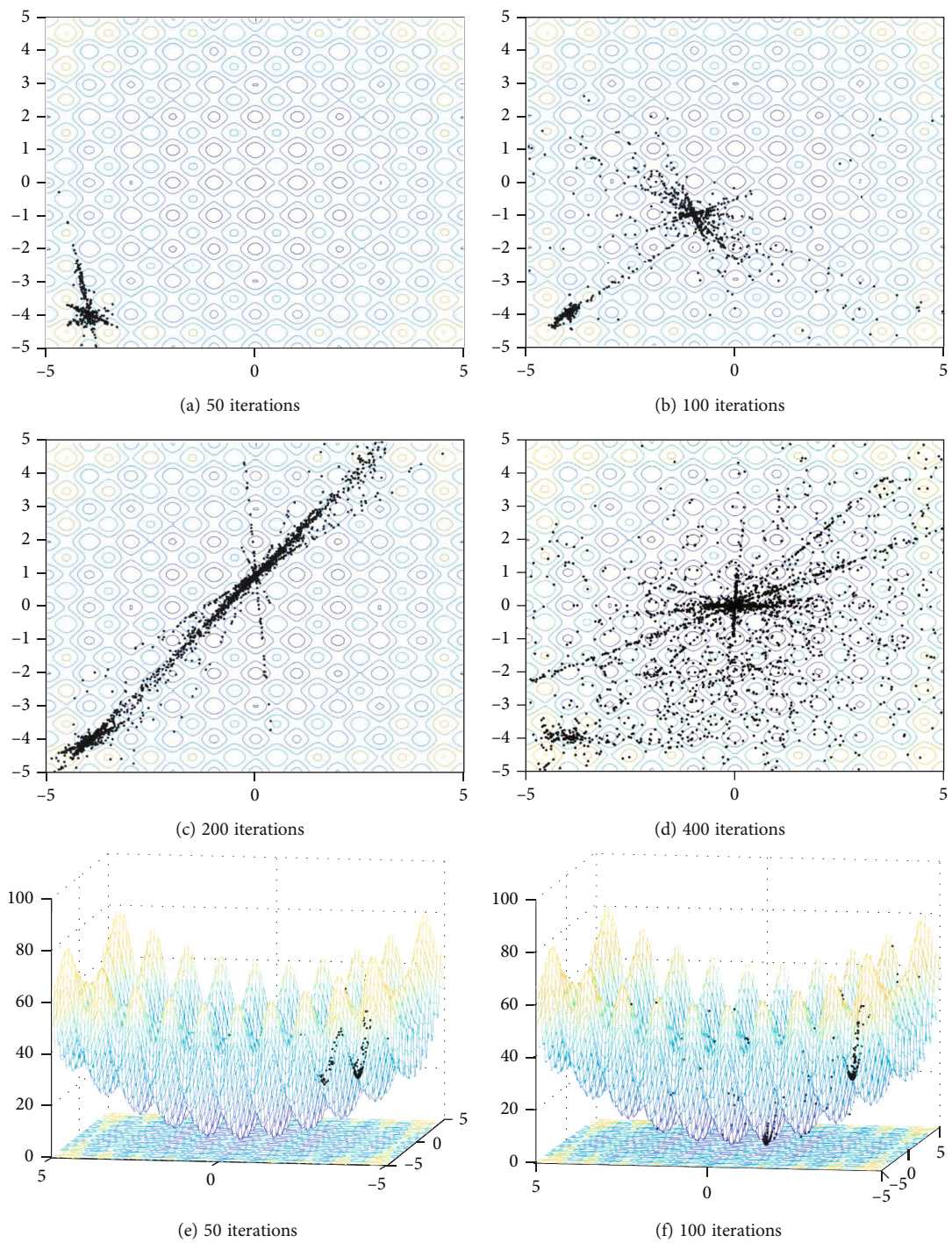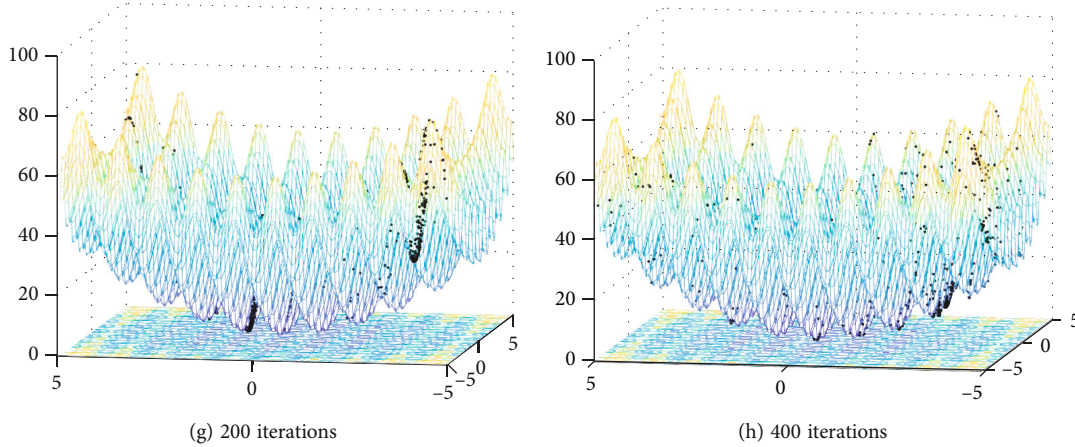
FIGURE 8: Behavior results of the point location history in FBS.

the fitness. We take the Fibonacci series as the total number of search nodes, and we need to save the most suitable F_{p+1} set of these points. Then, we need to remove other $3F_p - F_{p+1}$ nodes. The search space set in the pth iteration is updated from the saved points.

Corresponding to the global and local search stages, a schematic diagram of the Fibonacci branching process is shown in Figure 6.

The depth of the Fibonacci branching layer shown in Figure 6 is set to the expected value at the beginning, and the total number of nodes in each branching layer is stored in the Fibonacci sequence. In Figure 6, the white dotted circle is the search point set in the previous iteration process, the black solid border circle represents the endpoint of this iteration $\vec{x}_A$, and the gray real circle represents the global random endpoint $\vec{x}_B$. Figure 6(a) depicts the global search phase, which is the first stage of the whole process, in which the partition point $\vec{x}_{S1}$ represented by a white circle solid is made up based on the uniformly distributed points and $\vec{x}_A$. In Figure 6(b), other end nodes are fitted optimally in the local optimization stage $\vec{x}_A$ and $\vec{x}_B$ in the current iterative space is merged. Then, a new split node $\vec{x}_{S2}$ is obtained through the iterative rules. The adaptable values of $\vec{x}_A$, $\vec{x}_B$, $\vec{x}_{S1}$, and $\vec{x}_{S2}$ are assessed, and the optimal F_{p+1} solution of the objective function evaluation is preserved.

Figure 7 discloses a flow chart of a general process for a specific implementation of FBS.

3.3. Proof of Reachability and Convergence in FBS for Global Optimization of Multimodal Functions.
In this section, according to the properties of abovementioned Fibonacci, the reachability and convergence of Fibonacci are studied. Through strict mathematical proof, it is proven that the FBS-based algorithm proposed in this paper can determine the global optimal solution and ensure that the FBS algorithm converges to the optimal solution.

3.3.1. Accessibility Investigation of the FBS Algorithm.
Mathematical proof I FBS obtains the set of the solution objective function space by searching the reachable set in the space.

From the characteristics of the abovementioned algorithm, we know that after a sufficiently long iteration $n < +\infty$, rule 1 generates uniformly distributed endpoints $\vec{x}_B$ in a constrained space, such as $\forall \vec{X} \in \vec{x}_B, \vec{X} = (\vec{x}_d)_{D\times1}$. D are dimensions of points, their probability distribution is $P(\vec{x}_d) = U(\vec{x}_{\min}, \vec{x}_{\max})$, and $\vec{x}_d$ satisfies the following relationship of the objective function field:

$$\int \overset{D.}{\cdots} \int \frac{1}{\vec{x}_{\max} - \vec{x}_{\min}} d\vec{x}_d = 1. \tag{8}$$

The abovementioned theoretical proof process shows that $\forall \vec{X} \in \vec{x}_B$ obeys $\vec{X} \in B$; optimization set B is the reachable set of $\vec{x}_B$ obtained by the FBS algorithm.

In mathematical proof II, for the FBS algorithm, the overall situation optimality of the objective function in the search field is feasible.

Assume that the overall optimal solution $\vec{x}^*$ of the search field is in the field B. It is proved by theory that the solution $\vec{x}^*$ is in reachable B. After that, we assume that the probability in a uniform distribution is P and then assume that the result obtained by the algorithm is $\vec{x}^L$. On the basis of the proof, after a sufficiently long iteration, the probability of FBS reaching the saddle point is $P^* \leq \prod_{i=1}^{n}(1 - P)$ to obtain the result

$$\lim_{n \longrightarrow +\infty} P^* \leq \lim_{n \longrightarrow +\infty} \prod_{i=1}^{n}(1 - P) = 0. \tag{9}$$

From this point of view, the FBS algorithm can obtain the overall best solution of the objective function in the search space.

(a) 50 iterations

(b) 100 iterations

(c) 200 iterations

(d) 400 iterations

(e) 50 iterations

(f) 100 iterations

Figure 9: Continued.

(g) 200 iterations

(h) 400 iterations
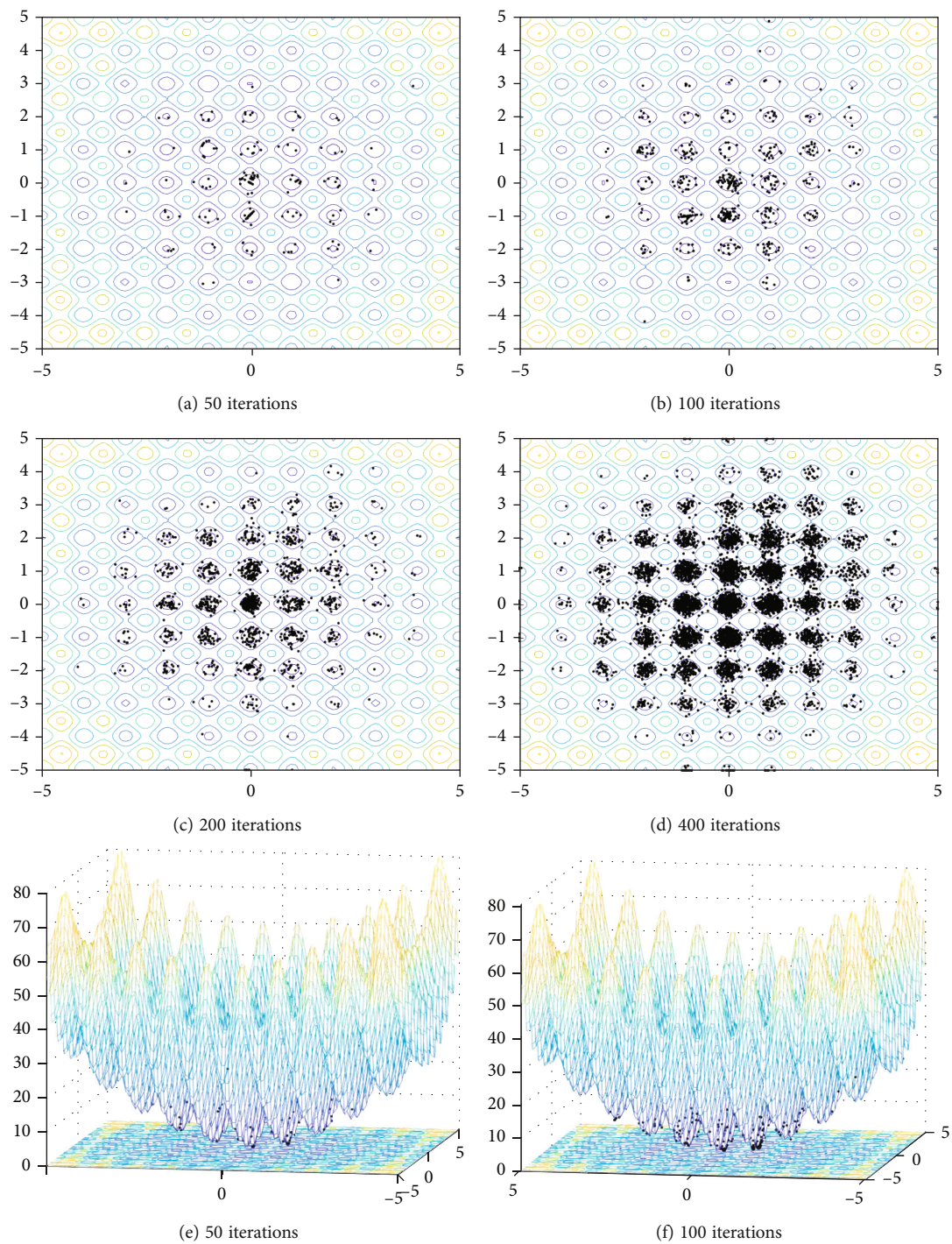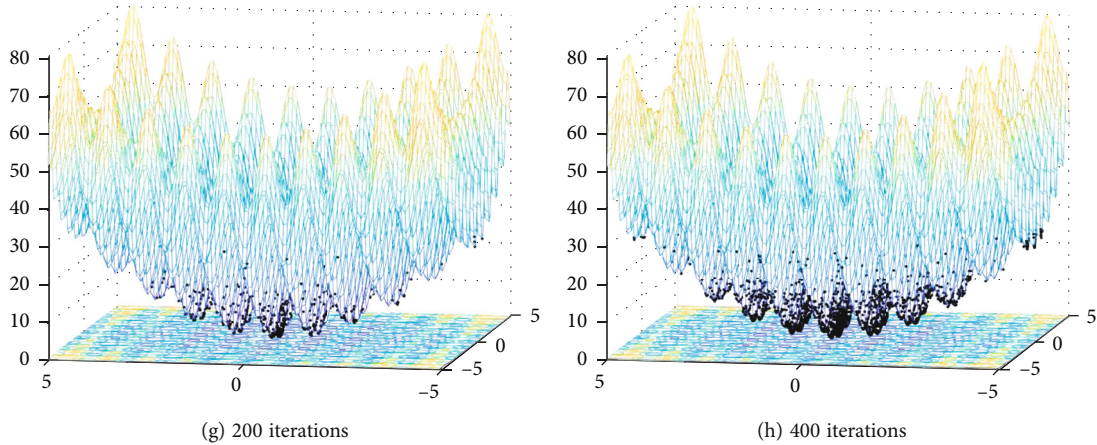
FIGURE 9: Behavior results of the point location history in GSA.

TABLE 2: Simulation experiment parameters.

Param.	Value
Size of area	$(1\,\text{km} \times 1\,\text{km})$
Number of sensors N_S	100
Transmitting range of sensors R_S	150 m
Jamming range of jammer R_J	150 m
Routing protocol	OSPF

Therefore, according to the abovementioned proofs, we can obtain that the overall optimal solution of the objective function field is accessible in the search field.

3.3.2. Convergence Analysis of the FBS Algorithm. Assuming that FBS achieves a sufficiently long iteration $n < +\infty$ to find the overall optimal solution, the existing G of the basic construction generated by FBS could become a gradually optimized node set $\{\vec{x}_T\} = \{\vec{x}_t | \vec{x}_{\text{best}} \in S_t\}, t = 1, 2, \cdots, n$. $\vec{x}_{\text{best}}$ is the best result of the iteration set. Afterwards, we construct the probability $P(t) = P(|\vec{X}^* - \vec{x}_t| \geq \varsigma)$ of FBS converging to the overall best result. ς indicates a small pinned variable. According to the rules $P(t) = P_{\overline{X}}(t) + P_{\varsigma}(t)$, $P_{\overline{X}}(t)$ is the chance of creating an equidistribution of random nodes of the field defined by rule 1, while $P_{\varsigma}(t)$ is the chance of creating an equidistribution of random nodes of the field defined by the radius parameter ς of rule 2. Thus, when there are t iterations of FBS, the search point does not reach the ς interval region around the global optimal solution, $\vec{X}^*$, $\tilde{P}(t) = P(|\vec{X}^* - \vec{x}_t| \geq \varsigma)$; then, $\lim_{t \to +\infty} \tilde{P}(t) = 0$ is obtained and the following results are obtained:

$$P(n<+\infty) > P(t) = 1 - \tilde{P}(t)P(n<+\infty) > 0. \quad (10)$$

Letting $t \longrightarrow +\infty$, we can obtain

$$P(n<+\infty) = 1. \quad (11)$$

It is proven that FBS converges to the overall best result with a probability of 100%.

Through the proof in Sections 3.3.1 and 3.3.2 above, the results show that the designed FBS algorithm is feasible and convergent and finally can obtain the overall best result.

4. Jammer Location-Aware Method Based on Fibonacci Branch Search

FBS-based jammer location awareness mainly includes three steps: a selection of initial search points, a clarification of the fitness function, and a search point update.

In the process of location awareness, to narrow the search scope, the search points should be all within the mapped area. In the pth iteration, we assume that F_p search points are selected in the mapped area randomly and the coordinates of the ith search point are $\mathbf{p}_i = [x_i^P, y_i^P], i = 1, 2, \cdots, F_p$.

To evaluate the performance of the estimation result of the jammer's location quantitatively, the fitness function is designed in this section. When a jamming attack is conducted, the jammed area is approximately a circle and the distance between the jammer and the farthest boundary node is approximately equal to the jamming radius. Based on the analysis result in Section 2.2, we can obtain the coordinates of the boundary nodes. Then, the distances between the boundary nodes and the search point are calculated as the fitness function.

In the pth iteration, there are F_p search points in the mapped area and the coordinates of the ith search point are denoted as $\mathbf{p}_i = [x_i^P, y_i^P], 1 \leq i \leq F_p$. Then, the fitness function at the pth iteration for the ith search point is denoted as

$$\text{Fit}_i(p) = \frac{1}{N_B} \sum_{j=1}^{N_B} |d_{ij}(p) - \bar{d}_i(p)|, \quad i = 1, 2, \cdots, N_B, \quad (12)$$

where N_B is the number of boundary nodes, $[x_j^B, y_j^B]$ are the coordinates of the jth boundary node, and $d_{ij}(p)$ is the distance between the ith search point and the jth

(a) The first of the iteration of the global search

(b) The first iteration of the local search

(c) The second iteration of the global search

(d) The second iteration of the local search
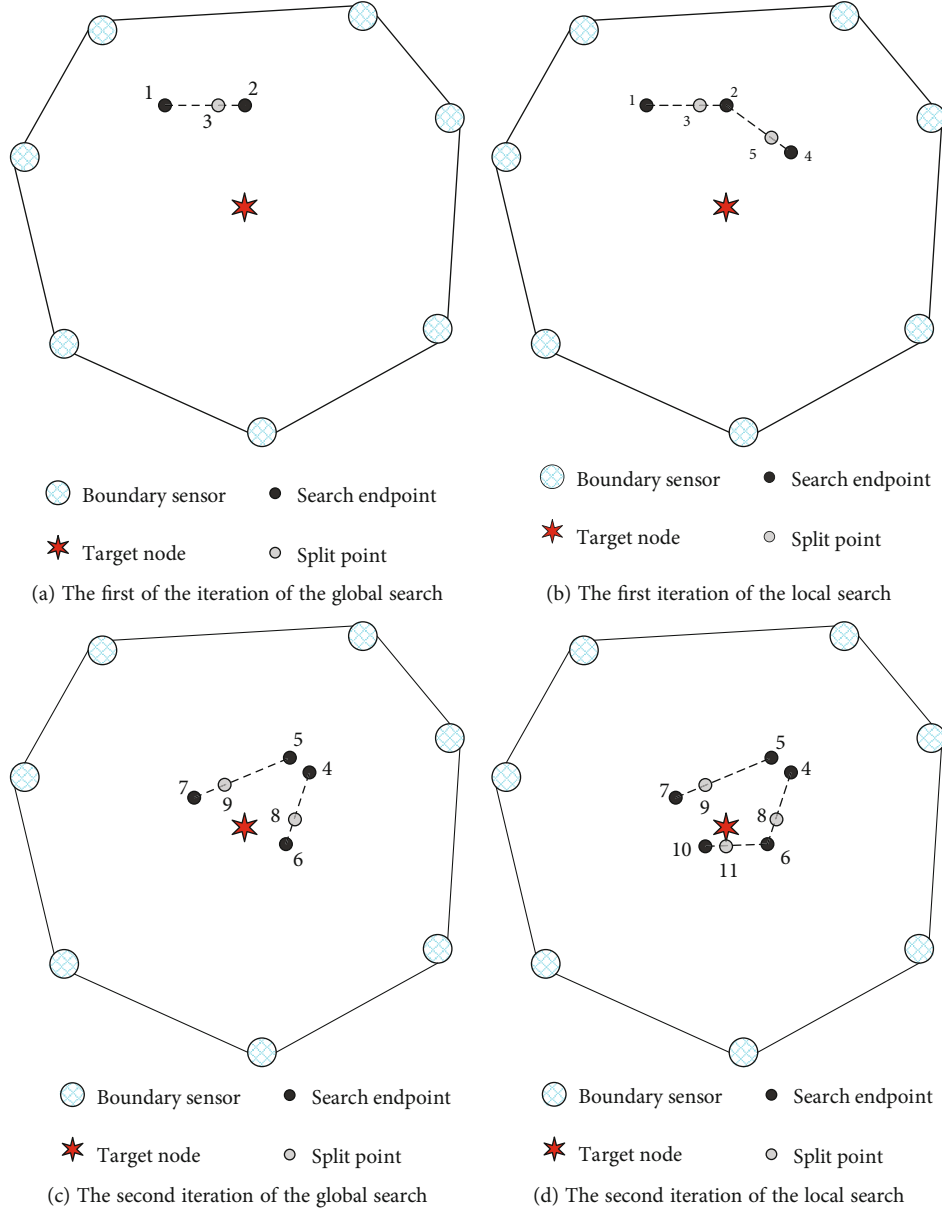
FIGURE 10: Schematic diagram of positioning process based on FBS.

boundary node. $\bar{d}_i(p)$ is the average distance between the ith search point and all of the boundary nodes. $d_{ij}(p)$ is calculated as

$$d_{ij}(t) = \sqrt{\left(x_i^P(p) - x_j^B\right)^2 + \left(y_i^P(p) - y_j^B\right)^2}, \quad (13)$$

and $\bar{d}_i(t)$ is calculated as

$$\bar{d}_i(t) = \frac{1}{N_B} \sum_{j=1}^{N_B} \left(d_{ij}(t)\right). \quad (14)$$

After the fitness value of each search point is calculated, all of these points are sorted in a descending order

from the best to the worst; the top F_{p+1} points are saved, while the F_{p+1} search points form a new set $\mathbf{G}_{p+1}$ for the next optimization.

In the process of location awareness, the goal is to search for the search point of the smallest fitness value. The pseudocode of the FBS-based location-aware algorithm is shown in Algorithm 1. The input is the boundary sensors' coordinate set $\mathbf{B}$, the branch depth R, the maximum iterations N, and the initial search point set $\mathbf{G}_p$, and the output is the estimated result of the jammer's coordinates $\hat{\mathbf{r}}$.

The coordinates of the search node with the best adaptable value are selected as the coordinates of the jammer which is represented as $\hat{\mathbf{r}}$.

The implementation of FBS is completed by generating search elements and search branches. Assuming that the
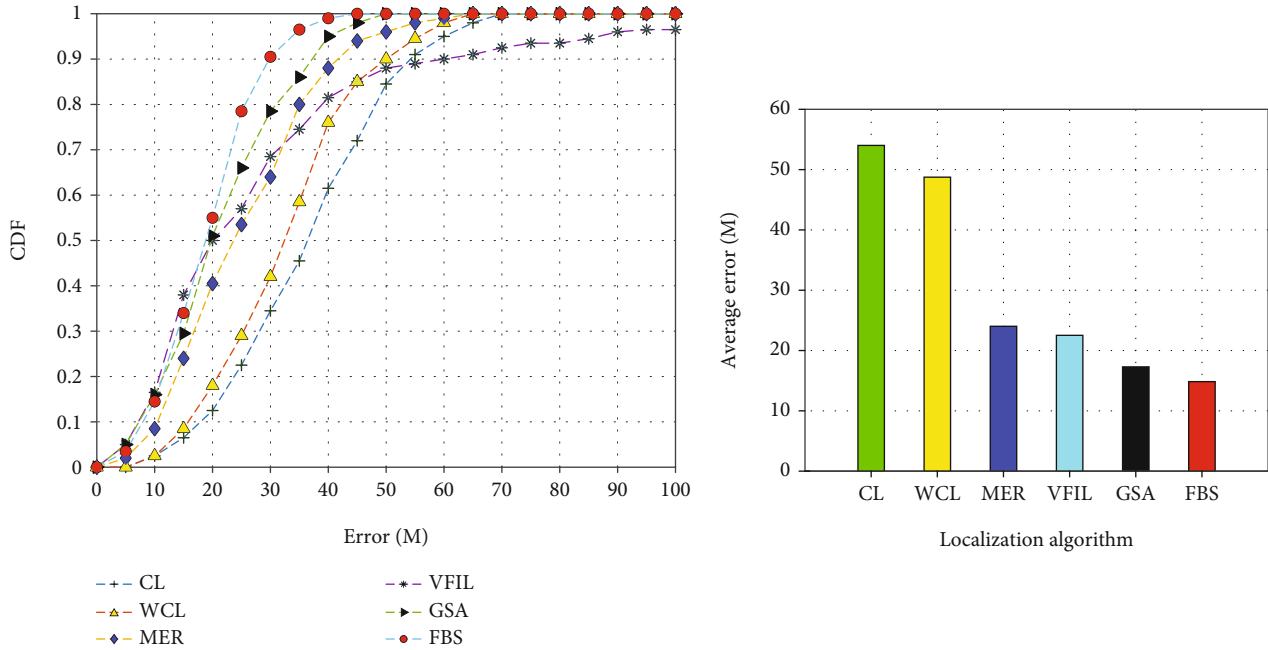
(a) The cumulative distribution functions of location-aware errors

(b) Average location-aware errors

FIGURE 11: Analysis of location-aware errors when the number of sensors $N_S = 100$ and the jamming range $R_J = 150$ m.

optimization space is considered N_e when FBS is used in location awareness of the jammer, the comparison dimension in the search element is set as N_c; then, the algorithm complexity of FBS is approximate $O(C_f N_e (N_c)^2)$, where $C_f = \sum_{i=1}^{N}(F_i)$ is the final sum of the searched elements in the Fibonacci search branch under the maximum number of loop iterations. Thus, it can be seen from the abovementioned FBS complexity analysis that the computational complexity of our proposed optimization algorithm mainly depends on the dimension of the search element and the maximum number of iterations.

5. Simulation Experiments

In this section, we analyze the performance of the proposed FBS algorithms using abundant computer simulations. The simulation is realized in EXata, which is an excellent simulator for wireless networks, and the analysis of the simulation result is realized in MATLAB.

5.1. Location History of the Search Points in FBS for the Rastrigin Function. In this part, the proposed edge-back method with global optimization capabilities proves that the use of location historical search points in the optimization iteration process finds the global optimal solution instead of falling into the benchmark of local optimization and is combined with the gravity search algorithm (GSA). The benchmark function selected in this part is the Rastrigin function, which has multiple local optimal solutions and a globally optimal solution.

　　The performance of the proposed fullback movement trajectory search points is scattered in the best solution,

and the search space for the optimal convergence point in Rastrigin is shown in Figure 8. This digital display of the fullback model can simulate the three-dimensional position of the historical search point and the trajectory profile in different iterations. To compare the performance of the FBS-based algorithm, we compared the algorithm with GSA and the result of GSA is shown in Figure 9. In FBS and GSA, the initial position of the search point is set at the extreme local optimal point.

　　As shown in Figures 8 and 9, the search points constantly explore the potential areas in the solution space and finally cluster around the global optimal value in a multimodal Rastrigin mode. The experiments displayed in Figure 9 show that with the increase of the number of iterations, the point cluster of the GSA algorithm is gradually in the extreme point and continues to maintain the local optimal conditions, with almost no particles searching for the global optimal extreme point. Based on a local optimum of the Rastrigin function, it is further proven that the algorithm is essentially trapped by the local optimum condition and falls into the local search space. It can be found under the same conditions from the search point trajectory and the 3D version shown in Figure 8. Although the Rastrigin function is asymmetric and multichannel and has different mountain levels, it is found that the global best challenge comes from many local variables. The minimum value is in the search space. It is worth noting that with the help of a global random search, FBS can be jumped out from the local optimal solution at the extreme point and from the notch solution at the local optimal point. Throughout the historical process of locating search points, in the two-dimensional and three-dimensional space iterations, the points converging to the global optimal condition and the

(a) $N_S = 200$

(b) $N_S = 300$
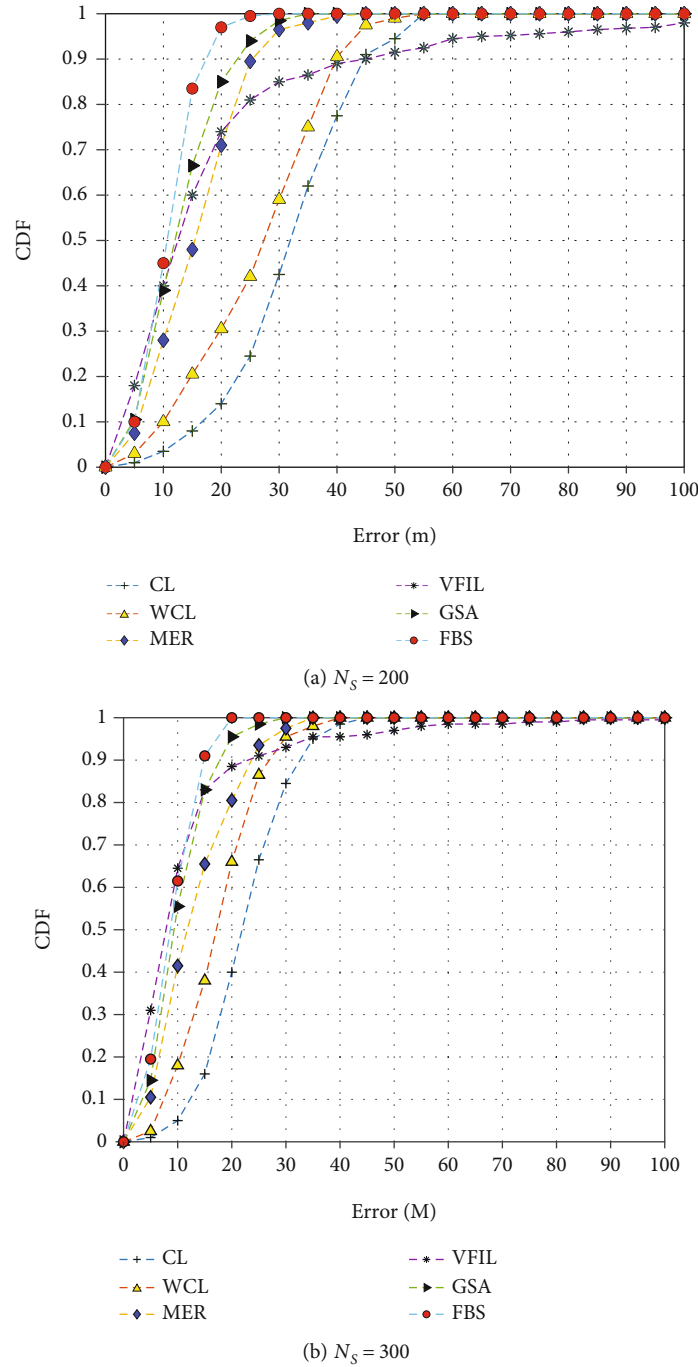
FIGURE 12: The CDF of location-aware errors when the number of sensors is different, and the jamming range $R_J = 150$ m.

regions initializing the optimal growth point are often scattered in the extreme points and gradually move to the optimal solution in the search space. After the first 50 iterations, more than half of the agents are close to the global optimal valley and begin to converge to the optimal valley. As the number of iterations increases, an increasing number of agents gather and disperse near the extreme point, especially in the global optimal target region. Finally, the search point finds the global optimum solution and converges to the global optimum, which can be investigated and reasoned by introducing the concept of global randomness into an endpoint generated by FBS rule 1. To ensure the convergence of the algorithm, local development and optimization capabilities are emphasized at other endpoints. Because the global random point performs a global search in space, it usually moves from a less suitable universe to a more suitable universe. The best universe will be saved and moved to the next search. Therefore, these capabilities and behaviors will help the FBS algorithm not fall into a local optimum and quickly converge to the optimal target point.

Simulation and discussion prove the effectiveness and convergence of the FBS algorithm. The algorithm proposed
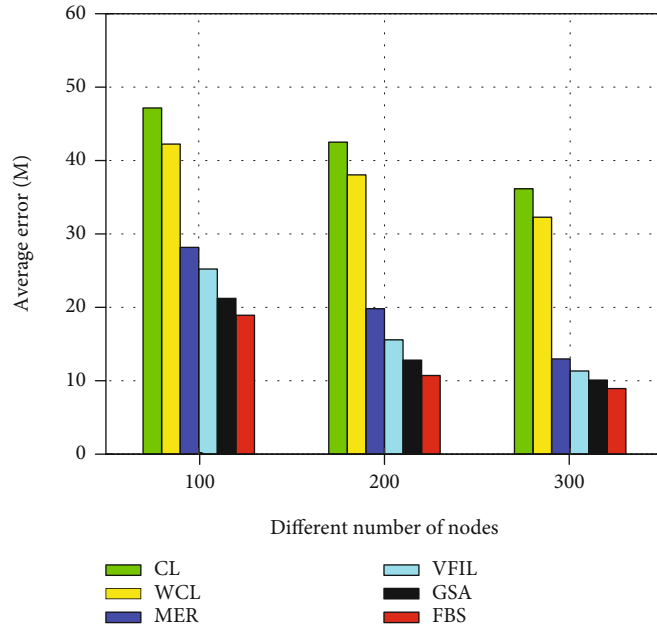
FIGURE 13: Influence of node density on the average location-aware errors of different algorithms.

in this article can find the global optimal solution in the solution space.

5.2. Parameters Setting and Benchmark in the Location-Aware Method. For every experiment, the sensors and the jammer are all randomly deployed in the (1 km × 1 km) area. The simulation experiment parameters of the sensors and the jammer are listed in Table 2.

To realize the quantitative analysis of location-aware results, the average location-aware error is utilized to evaluate the performance of each method, which is denoted by E and given by

$$E = \frac{\|\mathbf{r} - \hat{\mathbf{r}}\|}{N(E)}, \tag{15}$$

where $N(E)$ is the number of simulation experiments, $\mathbf{r}$ is the real location of the jammer, and $\hat{\mathbf{r}}$ is the jammer coordinate calculated by the algorithm. In addition, the cumulative distribution functions (CDFs) of the average location-aware error are also considered.

5.3. Performance Comparison and Result Analysis. Use the FBS-based noncooperative target node location algorithm to locate the communication node in the wireless sensor network and take Figure 10 as an example to describe the location process.

Figure 10(a) shows the global search stage during the first iteration. First, randomly select a search point (node 1) in the mapping area as the search endpoint. Then, randomly select a node (node 2) as the search endpoint and obtain the split point (node 3) according to the calculation formula of the split point. The fitness values of node 1, node 2, and node 3 are calculated and sorted, and the fitness value of node 2 is the best,

which is used as the endpoint for local optimization. As shown in Figure 10(b), randomly select a search point (node 4) as the endpoint, find the split point (node 5) according to criterion 2, sort the fitness values of all nodes, and select the best two nodes (node 4 and node 5) used as the endpoints of the second iteration. In the global search process of the second iteration of Figure 10(c), two search points (node 6 and node 7) are randomly selected and the split point between node 4 and node 6 (node 8). The dividing point between node 5 and node 7 (node 9). Calculate and sort the fitness values of all nodes in the area. In the second local search stage of Figure 10(d), node 6 with the best fitness value is selected as the endpoint. Randomly select the search point (node 10) as another search point, and find the split point between node 6 and node 10 (node 11). Calculate and sort the fitness values of node 4, node 5, node 6, node 7, node 8, node 9, node 10, and node 11, and select node 6, node 10, and node 11 as the search for the third-iteration endpoint.

For every scenario, 10^3 experiments are conducted, and in every scenario, we compare the performance of FBS with CL, WCL, MER, VFIL, and GSA. The CDF of the average location-aware error when $N_S = 100$, $R_J = 150$ m is shown in Figure 11(a). Figure 11(b) shows the average location-aware errors of different location-aware algorithms. As we can see from the results, the error of the FBS-based location-aware algorithm is lower than those of the other algorithms.

Assuming that the area size is constant, the sensor number can be used to reflect the sensor density in the area. To analyze the influence of different node densities on the performance of different algorithms when the jamming range of the jammer is set as 150 m, the number of sensors is set as 200 and 300. The CDF of location-aware errors for different node densities is presented in Figure 12 after conducting 10^3 experiments independently. Figure 12(a) presents the CDF

(a) Jamming range $R_J = 200\,\mathrm{m}$

(b) Jamming range $R_J = 250\,\mathrm{m}$
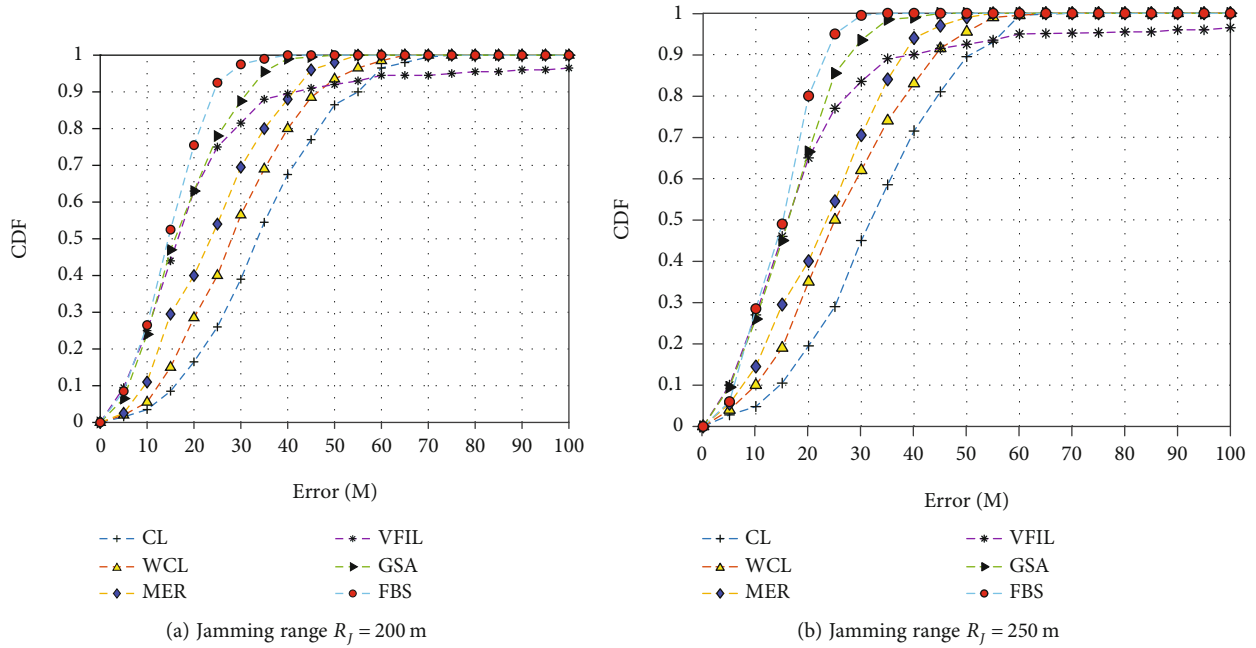
FIGURE 14: CDF of location-aware errors when the number of sensors $N_S = 100$; the jamming range is different.
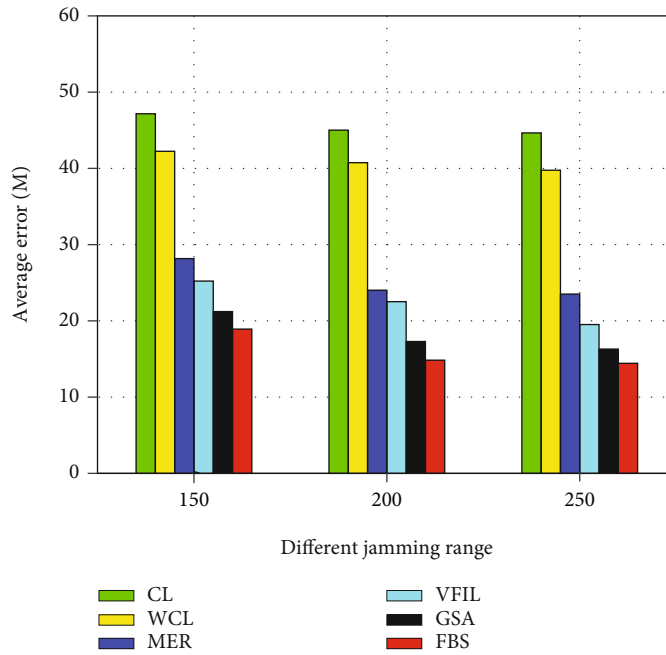
FIGURE 15: Influence of the jamming range on average location-aware errors of different algorithms.

of location-aware errors when $N_S = 200$, and Figure 12(b) presents the CDF of location-aware errors when $N_S = 300$. The average location-aware error change with different node densities is shown in Figure 13. As we can find in Figure 13, the average location-aware error of FBS is smaller than other algorithms, which means that the performance of FBS is better than the other algorithms. CL, WCL, MER, and VFIL are more sensitive to the distribution of sensors than GSA and FBS.

To analyze the influence of different jamming ranges on the performance of different algorithms, the number of sensors in the area is set as 100 and the jamming ranges are set as 150 m, 200 m, and 250 m. After conducting 10^3 experiments independently, the CDF of location-aware errors for different jamming ranges is presented in Figure 14. Figure 14(a) presents the CDF of location-aware errors when $R_J = 200\,\mathrm{m}$, and Figure 14(b) presents the CDF of location-aware errors when $R_J = 250\,\mathrm{m}$. The average location-aware

error change with different jamming ranges is shown in Figure 14(b). As we can see in Figure 15, the average location-aware error of FBS is smaller than that of the other algorithms, which illustrates the better performance of FBS; when R_J = 250 m, the average location-aware error of FBS is approximately 15 m.

6. Conclusion and Future Work

Predicting the active location of jammers and removing them can ensure the safety of WSNs. This paper proposes an evolutionary algorithm FBS based on the Fibonacci sequence method, which reduces the sensitivity of existing algorithms to the deployment and parameters of WSN jammer location awareness. Although this FBS has been studied in part of the literature, the accessibility and convergence of the FBS algorithm have not been proven. In this paper, the reachability and convergence of FBS are strictly proven, which further verifies the validity of the theory and supports the previous view. In the process of jammer location awareness, the boundary sensors in the jammed area are identified by the map service of the jammed area and the fitness function is constructed by the distance from the search point to all boundary sensors. After iteration, the position of the best fitness search point is estimated as the position of the jammer. The experimental results are compared with the Cl, WCL, mer, vfil, and GSA algorithms. The experimental results show that the location-aware algorithm based on FBS has a good performance, and the location-aware result is more accurate than that of the other algorithms.

In the real world, the jammer may be in a mobile state all the time, which increases the difficulty of positioning the jammer. At the same time, wireless sensor networks have hardware limitations on battery power due to cost; complex jammer location-aware algorithms will consume a lot of energy. Therefore, designing a simple and efficient algorithm that can get awareness of the location of mobile jammers is our future research direction.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] T. Wang, X. Wei, J. Fan, and T. Liang, "Jammer localization in multihop wireless networks based on gravitational search," *Security and Communication Networks*, vol. 2018, 11 pages, 2018.

[2] L. Chhaya, P. Sharma, G. Bhagwatikar, and A. Kumar, "Wireless sensor network based smart grid communications: cyber attacks, intrusion detection system and topology control," *Electronics*, vol. 6, no. 1, p. 5, 2017.

[3] A. Gaddam, T. Wilkin, M. Angelova, and J. Gaddam, "Detecting sensor faults, anomalies and outliers in the internet of things: a survey on the challenges and solutions," *Electronics*, vol. 9, no. 3, p. 511, 2020.

[4] S. Misra, R. Singh, and S. V. R. Mohan, "Information warfare-worthy jamming attack detection mechanism for wireless sensor networks using a fuzzy inference system," *Sensors*, vol. 10, no. 4, pp. 3444–3479, 2010.

[5] W. Xu, K. Ma, W. Trappe, and Y. Zhang, "Jamming sensor networks: attack and defense strategies," *IEEE Network*, vol. 20, no. 3, pp. 41–47, 2006.

[6] S. Vadlamani, B. Eksioglu, H. R. Medal, and A. Nandi, "Jamming attacks on wireless networks: a taxonomic survey," *International Journal of Production Economics*, vol. 172, pp. 76–94, 2016.

[7] L. Ye, J. Fulong, L. Hao, W. Jianhui, H. Chen, and Z. Meng, "Interference robust channel hopping strategies for wireless sensor networks," *China Communications*, vol. 13, no. 3, pp. 96–104, 2016.

[8] M. S. Al-kahtani, L. Karim, and N. Khan, "Efficient opportunistic routing protocol for sensor network in emergency applications," *Electronics*, vol. 9, no. 3, p. 455, 2020.

[9] D. Ciuonzo, A. Aubry, and V. Carotenuto, "Rician MIMO channel- and jamming-aware decision fusion," *IEEE Transactions on Signal Processing*, vol. 65, no. 15, pp. 3866–3880, 2017.

[10] Y. Zou, J. Zhu, X. Wang, and L. Hanzo, "A survey on wireless security: technical challenges, recent advances, and future trends," *Recent Advances, and Future Trends*, vol. 104, no. 9, pp. 1727–1765, 2016.

[11] T. Wang, X. Wei, J. Fan, and T. Liang, "Adaptive jammer localization in wireless networks," *Computer Networks*, vol. 141, pp. 17–30, 2018.

[12] A. Shoari and A. Seyedi, "Localization of an uncooperative target with binary observations," in *2010 IEEE 11th International Workshop on Signal Processing Advances in Wireless Communications*, pp. 1–5, Marrakech, Morocco, June 2010.

[13] X. Wei, Q. Wang, T. Wang, and J. Fan, "Jammer localization in multi-hop wireless network: a comprehensive survey," *IEEE Communication Surveys and Tutorials*, vol. 19, no. 2, pp. 765–799, 2017.

[14] F. Jameel, M. A. A. Haider, and A. A. Butt, "Robust localization in wireless sensor networks using RSSI," in *2017 13th International Conference on Emerging Technologies*, pp. 1–6, Islamabad, Pakistan, December 2017.

[15] B. Jin, X. Xu, and T. Zhang, "Robust time-difference-of-arrival (TDOA) localization using weighted least squares with cone tangent plane constraint," *Sensors*, vol. 18, no. 3, pp. 1–16, 2018.

[16] Y. Cong, X. Sun, and M. Sun, "Refinement of TOA localization with sensor position uncertainty in closed-form," *Sensors*, vol. 20, no. 2, pp. 1–19, 2020.

[17] S. C. K. Herath and P. N. Pathirana, "Optimal sensor arrangements in angle of arrival (AoA) and range based localization with linear sensor arrays," *Sensors*, vol. 13, no. 9, pp. 12277–12294, 2013.

[18] N. S. Ewa, "Localization in wireless sensor networks: classification and evaluation of techniques," *International Journal of Applied Mathematics and Computer Science*, vol. 22, no. 2, pp. 281–297, 2012.

[19] F. Thomas and L. Ros, "Revisiting trilateration for robot localization," *IEEE Education Society*, vol. 21, no. 1, pp. 93–101, 2005.

[20] I. Borg and P. J. Groenen, "Modern multidimensional scaling: theory and applications," *Journal of Educational Measurement*, vol. 40, no. 3, pp. 277–280, 2003.

[21] Y. Shang, W. Ruml, Y. Zhang, and M. P. Fromherz, "Localization from mere connectivity," in *Proceedings of the 4th ACM international symposium on Mobile ad hoc networking & computing*, pp. 201–212, Annapolis, Maryland, USA, June 2003.

[22] Y. Shang and W. Ruml, "Improved MDS-based localization," in *In Proceedings of the 2004 IEEE International Conference on Computer Communications (INFOCOM)*, pp. 2640–2651, Hong Kong, China, March 2004.

[23] G. T. F. de Abreu and G. S. Destino, "Super MDS: source location from distance and angle information," in *In Proceedings of the 2007 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 4430–4434, Hong Kong, China, March 2007.

[24] M. Amirinasab Nasab, S. Shamshirband, A. T. Chronopoulos, A. Mosavi, and N. Nabipour, "Energy-efficient method for wireless sensor networks low-power radio operation in Internet of things," *Electronics*, vol. 9, no. 2, p. 320, 2020.

[25] A. Shoari and A. Seyedi, "Target localization with binary observations: effect of censoring non-detecting sensors," in *In Proceedings of the 2011 IEEE International Symposium on Information Theory Proceedings*, pp. 2504–2508, St. Petersburg, Russia, August 2011.

[26] A. D. Wood, J. A. Stankovic, and S. H. J. A. M. Son, "A jammed-area mapping service for sensor networks," in *In Proceedings of the 24th IEEE International Real-Time System Symposium (RTSS)*, pp. 286–297, Cancun, Mexico, December 2003.

[27] W. Xu, W. Trappe, Y. Zhang et al., "The feasibility of launching and detecting jamming attacks in wireless networks," in *In Proceedings of the 6th ACM International Symposium on Mobile Ad Hoc Networking and Computing*, pp. 46–57, Urbana-Champaign, Illinois, USA, May 2005.

[28] N. Bulusu, V. Bychkovskiy, D. Estrin, and J. Heidemann, "Scalable, ad hoc deployable RF-based localization," in *In Proceedings of the Grace Hopper Conference on Celebration in Women Computer*, pp. 1–5, Vancouver, Canada, Octobor 2002.

[29] J. Blumenthal, R. Grossmann, F. Golatowski, and D. Timmermann, "Weighted centroid localization in zigbee-based sensor networks," in *In Proceedings of the 2007 IEEE International Symposium on Intelligent Signal Processing (WISP)*, pp. 1–6, Alcala de Henares, Spain, Octobor 2007.

[30] H. Liu, X. Wenyuan, Y. Chen, and Z. Liu, "Localizing jammers in wireless networks," in *In Proceedings of the 2009 IEEE International Conference on Pervasive Computing and Communications*, pp. 1–6, Galveston, TX, USA, March 2009.

[31] T. Cheng, P. Li, S. Zhu, and D. Torrieri, "M-cluster and X-ray: two methods for multi-jammer localization in wireless sensor networks," *Integrated Computer-Aided Engineering*, vol. 21, no. 1, pp. 19–34, 2014.

[32] Q. Wang, J. Fan, X. Wei, and T. Wang, "Multi-jammers localization formulti-hop wireless network," *The Journal of Communication*, vol. 37, no. 13, pp. 176–186, 2017.

[33] B. Yildiz and E. Karaduman, "On Fibonacci search method with _k_ -Lucas numbers," *Applied Mathematics and Computation*, vol. 143, no. 2-3, pp. 523–531, 2003.

[34] A. Etminaniesfahani, A. Ghanbarzadeh, and Z. Marashi, "Fibonacci indicator algorithm: a novel tool for complex optimization problems," *Engineering Applications of Artificial Intelligence*, vol. 74, pp. 1–9, 2018.

[35] M. Subasi, N. Yildirim, and B. Yildiz, "An improvement on Fibonacci search method in optimization theory," *Applied Mathematics and Computation*, vol. 147, no. 3, pp. 893–901, 2004.

[36] J. O. Omolehin, M. A. Ibiejugba, A. E. Onachi, and D. J. Evans, "A Fibonacci search technique for a class of multivariable functions and ODEs," *International Journal of Computer Mathematics*, vol. 82, no. 12, pp. 1505–1524, 2005.

[37] X. Wang, D. J. Lyu, and Y. Dong, "Cutting parameters multischeme optimization based on Fibonacci tree optimization algorithm," *Control and Decision*, vol. 33, pp. 1373–1381, 2018.

[38] H. Zhang and F. Zeng, "Implementation of a novel Fibonacci branch search optimizer for the design of the low sidelobe and deep nulling adaptive beamformer," *International Journal of Microwave and Wireless Technologies*, vol. 12, no. 7, pp. 660–677, 2020.

Research Article

SVM Classification Method of Waxy Corn Seeds with Different Vitality Levels Based on Hyperspectral Imaging

Jinghua Wang [iD],[1,2] Lei Yan [iD],[1] Fan Wang,[1] and Shanshan Qi [iD][3]

[1]School of Technology, Beijing Forestry University, Key Lab of State Forestry Administration for Forestry Equipment and Automation, Beijing 100086, China
[2]National Engineering Laboratory for Agri-Product Quality Traceability, Beijing 100048, China
[3]Bureau of ecology and environment of hanging district, No.35 Minzhu street, Weifang, Shandong 261100, China

Correspondence should be addressed to Lei Yan; mark_yanlei@bjfu.edu.cn

The vitality of corn seeds is a significant indicator for assessing the quality and yield of crops. In recent years, numerous information technologies have been adopted to analyze the seed vitality and provide support for efficient equipment. However, there are still some shortcomings in these technologies, which decrease the accuracy of identifying the seed vitality for various practical applications. In this paper, a synthesized classification method for seed vitality was proposed based on multisensor hyperspectral imaging. Firstly, hyperspectral images in the range of 370-1042 nm were collected for waxy corn seeds, which were subjected to aging processing with four periods of time (0, 3, 6, and 9 d). Besides, some preprocessing techniques including standard normal variate, multiplicative scatter correction, Savitzky-Golay smoothing, and first-order and second-order derivatives were employed to suppress noise interference in raw spectra. In addition, principal component analysis (PCA), 2nd derivatization, and successive projection algorithm (SPA) were adopted to select feature wavelengths. Moreover, SVM classification models based on full spectra and feature wavelengths were established. The results showed that, based on feature wavelengths selected by SPA, the SVM model preprocessed by multiplicative scatter correction (MSC) had the optimal performance. The training accuracy and testing accuracy of this model were 100% and 97.9167%, respectively. RMSE was 0.018 and R^2 was 0.875. Therefore, it can be demonstrated that the pattern recognition algorithm could achieve a high accuracy in classifying accelerated aging seeds. This algorithm provides a new method for machine learning (ML) in nondestructive detection of crops.

1. Introduction

Seed vitality is one of the most important parameters that is directly related to seed germination performance and seedling emergence [1]. A suitable method for seed vitality detection can help farmers and seed companies reduce the deficit and favorably engage in agricultural production activities. The traditional method mainly relies on a manual germination test to distinguish seed vitality. It is time-consuming, inefficient, and inaccurate. Therefore, there is an urgent demand for a rapid and high-accuracy method for seed vitality detection.

The methods for seed vitality classification mainly contain chemical/biological methods and hyperspectral imaging

methods, apart from the manual germination test. Mcdonough et al. studied the vitality change trend of corn, sorghum, and sorghum flour under different aging grades by detecting the biological features of seeds and the chemical composition obtained by gel chromatography [2]. Cheyed measured the activities of amylase, phospholipase, protease, and phytase and explored the viability of wheat seeds in different storage periods [3]. The hydrogen peroxide (H_2O_2), ascorbic acid, and activity of catalase were determined to distinguish the stigma vitality of the rice [4]. RNA sequencing and DNA affinity purification sequencing analysis were performed to probe into the molecular mechanism of the rice seed germination [5]. Wei et al. investigated the protein and ultrastructure of the cotyledon and embryo, in an

attempt to classify the quality of different soybean seeds [6]. Although these methods are objective and accurate, they have such common disadvantages as heavy workload, inefficiency, and high professional requirements. Therefore, it can be hypothesized that they cannot detect the seed vitality efficiently.

Hyperspectral imaging is an emerging technique that integrates both spectroscopic and imaging techniques into one system [7]. It can be employed to reflect the internal information of seeds [8]. In recent years, it has been widely used in seed detection and has achieved excellent results. Yang et al. built a predicted model to detect whether the sugar beet seeds can germinate based on hyperspectral reflectance [9]. Şentaş et al. conducted an investigation into the yield of soybean (Glycine max) seeds based on the hyperspectral reflectance, which had favorable robustness [10]. Zhang et al. constructed models with spectral data to quickly, nondestructively, and accurately determine the germinated power of seeds [11]. Dumont et al. evaluated the seed quality of Norway spruce. They divided these seeds into three categories based on sparse logistic regression feature selection, and the accuracy of spectrum measurement reached 99% [12]. Hyperspectral technology provides a new way for rapid and nondestructive detection of seeds. However, each sample in the hyperspectral remote-sensing image has high-dimensional features and contains rich spatial and spectral information, which dramatically increase the difficulty of feature selection and mining [13]. With the advancement of artificial intelligence (AI), intelligence has been considered as the major challenge in promoting the economic potential and production efficiency of precision agriculture [14].

As an AI method, ML can effectively solve the problem of hyperspectral information feature selection and mining. Since the 1990s, AI has received extensive attention and has been adopted as a new learning method. The purpose is to determine the rules contained in a series of known samples, so that the machine can acquire a certain self-learning ability for unknown samples. A learning method, named support vector machine (SVM), has been developed on the basis of ML theory. It is a novel small sample learning method and can avoid the traditional process from "induction" to "deduction." SVM could simplify the usual classification problems and has presented multiple advantages over existing methods. Scholars maintain that SVM will strongly promote the development of ML theories and technologies [15].

SVM was the first classifier developed from the generalized portrait algorithm in pattern recognition. It was proposed by Soviet scholars Vladimir N. Vapnik and Alexander Y. Lerner in 1963 [16]. With the progression of theoretical research, SVM is gradually theorized and becomes a part of statistical learning theory. After decades of technical accumulation, SVM has been extensively applied in the field of classification and regression, including portrait recognition, text classification, handwritten character recognition, and bioinformatics.

In recent years, SVM has also been widely used in seed detection and has achieved excellent results. Baek et al. developed an SVM model to detect those rice seeds stained with lesions. The results showed that it was feasible to screen diseased rice seeds based on ML algorithms and spectral imaging technology [17]. Pattern recognition technology and data mining methods have become hotspots in chemometrics. SVM has been employed to classify different corn seeds based on spectral data. It has a high classification accuracy, which demonstrates the effectiveness of this method [18]. Despite the fact that significant efforts have been made to conduct investigations with respect to precision agriculture in previous studies, there remains a lack of a mature detection method for corn (Jingke 2000), an important economic crop, based on hyperspectral technology. Hence, the main purpose of this study is to explore the utilization of the SVM method to achieve rapid detection of the vitality of waxy corn seeds under different aging degrees. The main research contents are elucidated as follows.

(1) Elaborate on the relationship between seed vitality level and artificial aging time through standard germination tests, which could provide an experimental basis for the model construction

(2) Obtain the hyperspectral data of waxy corn seeds under different accelerated aging periods of time and use five preprocessing methods, including S-G smoothing, MSC, and SNV

(3) Adopt PCA and two other methods to filter feature wavelengths for the subsequent classification models

(4) Construct and compare SVM classification models under different pretreatment and feature selection methods, in an attempt to select the optimal model to identify waxy corn seeds

2. Related Works

The ML methods can be applied to the processing of hyperspectral imaging data, so as to realize the detection of seed vitality. Puneet et al. compared data visualization methods, such as principal component analysis (PCA) with multidimensional scaling (MDS), etc. They divided six kinds of tea into three different processing degrees according to the near-infrared spectral information [19]. Baek et al. tested several spectral preprocessing methods, such as continuous wavelet transform and feature selection methods, and improved the predicted accuracy by partial least square regression (PLSR) [20]. Wang et al. proposed an orthogonal signal correction (OSC) method for noise reduction and applied SPA to select the optimal wavelengths. A favorable PLSR model was obtained and can be employed to predict seed hardness based on hyperspectral imaging [21]. Insuck et al. used variable importance in projection (VIP) to remove redundant information and reduce the computation time for data processing. Two kinds of soybeans were classified by partial least square discrimination analysis (PLS-DA) based on spectra, which was confirmed to be feasible and effective [22]. These researchers continue to optimize the PLS model; for example, the PLS model processed by the orthogonal signal correction (OSC) method reduces the calculation factor
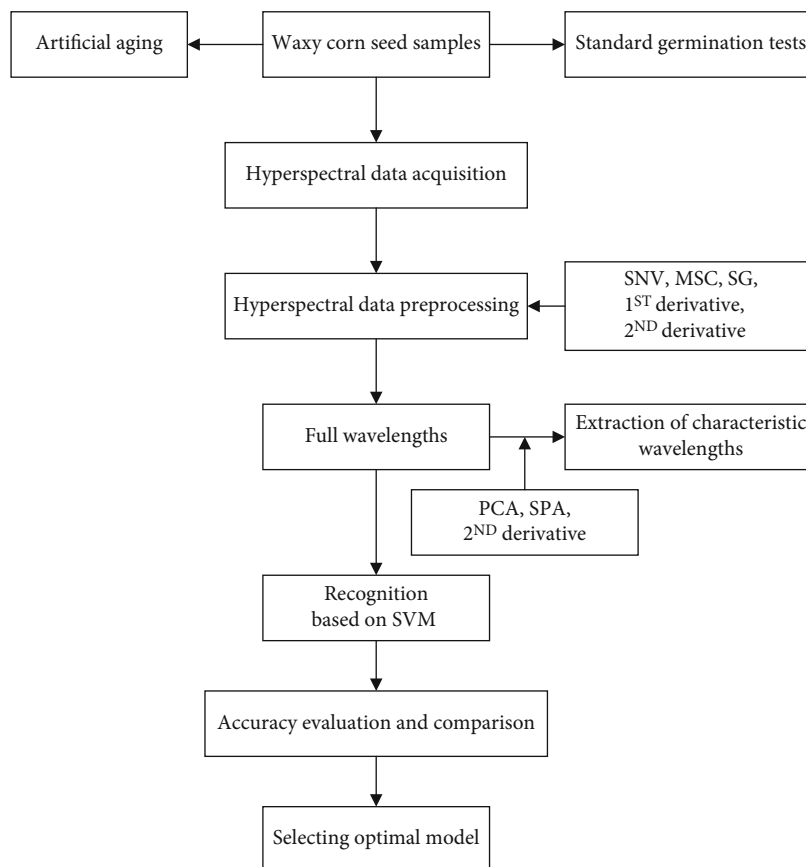
FIGURE 1: Flowchart for detection of waxy corn seed vitality based on SVM and hyperspectral imaging.

and improves the accuracy. Meanwhile, the PLS model processed by the variable importance in projection (VIP) method removes redundant information and improves the efficiency of the model. Nevertheless, the accuracy of the model is not significantly improved. Additionally, the PLS method is more suitable for the construction of a linear model, and it cannot accurately simulate the nonlinear relationship between seed vitality and spectra. However, the SVM model based on finding the optimal hyperplane for the feature space division is more suitable for analyzing the correlation between seed vitality and spectra. There are only a few support vectors that can be adopted to determine the classification results, due to the fact that they can grasp the key samples and remove the majority of redundant samples. The SVM method is characterized by high usability and better "robustness." Therefore, the SVM model was adopted in this study to classify the waxy corn seeds with different vitality levels based on hyperspectral imaging. Further, some other parameters were also analyzed.

3. Materials and Methods

The detailed flowchart of using SVM to detect the seed vitality of waxy corn is shown in Figure 1. The ultimate goal is to select the optimal SVM model combined with other pretreatment methods to detect seed vitality rapidly.

3.1. Modeling Methods

3.1.1. Preprocess Methods. After selecting the region of interest (ROI) of all corn seeds, the spectral data were preprocessed with five methods: MSC, SNV, S-G smoothing, 1st derivative, and 2nd derivative.

(1) Multiplicative Scatter Correction (MSC). MSC can be employed to compare the difference between the ideal spectra and the actual one [23]. It uses linear regression to achieve baseline correction and drift correction, which can correct the effects of scattering.

The basic steps are presented as follows.

(1) Average spectra calculation:

$$\bar{X} = \sum_{i=1}^{n} \frac{X_i}{n}.$$ (1)

(2) Linear regression analysis:

$$X_i = m_i \bar{X} + b_i,$$ (2)

where X represents the matrix of all spectral data, X_i represents the spectral reflectance of the i-th sample,
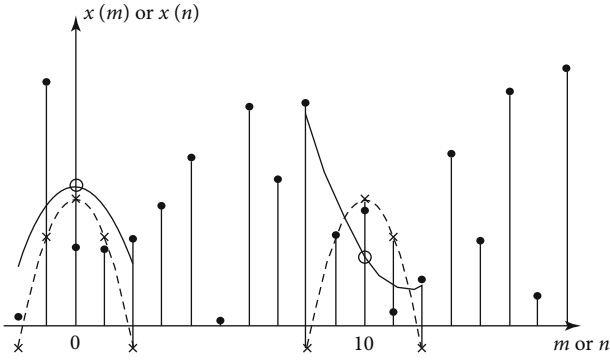
FIGURE 2: Schematic diagram of S-G smoothing.

and m_i and b_i represent the slope and intercept parameters, respectively between X_i and the average spectra.

(3) Correction using slope m_i and intercept b_i:

$$X_{i(\text{MSC})} = \frac{X_i - b_i}{m_i}, \tag{3}$$

where $X_{i(\text{MSC})}$ represents the spectral matrix after multiplicative scatter correction. This method can be employed to successfully correct all spectra and reduce the effects of scattering.

(2) Standard Normal Variate (SNV). SNV is a process in which each spectral curve is processed by standard normal transformation. It can weaken such interfering effects as scattering effects and light path changes [23]. It is more suitable for processing spectral data with large differences in samples. It can be assumed that in each spectrum, the spectral absorption value of each wavelength satisfies some conditions, such as a normal distribution; SNV can be employed to perform standard normal transformation processing on each spectrum, namely,

$$I_{\text{SNV}} = \frac{I - \mu}{\sigma}, \tag{4}$$

where I represents the original spectra value, μ represents the average value of the spectra, and σ represents the standard deviation of the raw data.

(3) Savitzky-Golay Smoothing. S-G smoothing is a filtering method based on local polynomial least square fitting to eliminate high-frequency random errors. Its prominent feature is that it can keep the shape and width of the signal unchanged while filtering the noise [24]. The basic theory can be illustrated by Figure 2.

A column of data $x[n]$ is represented by solid dots in the figure. A set of $2M + 1$ data can be considered to be centered on $n = 0$. It can be fitted with the following polynomial:

$$P(n) = \sum_{k=0}^{N} a_k n^k. \tag{5}$$

The residual of the least square fit is

$$\varepsilon_N = \sum_{n=-M}^{M} (p(n) - x[n])^2 = \sum_{n=-M}^{M} \left(\sum_{k=0}^{N} a_k n^k - x[n] \right)^2. \tag{6}$$

Only the constant term of the fitted polynomial needs to be obtained. It can be realized by convolution operation:

$$y[n] = \sum_{m=-M}^{M} h[m]x[n-m] = \sum_{m=n-M}^{n+M} h[n-m]x[m]. \tag{7}$$

(4) Derivative Analysis. The 1st-derivative and 2nd-derivative analyses can be adopted to reduce baseline correction and smooth background interference. It provides higher resolution and clearer spectral profile changes than the original spectra. The basic principle is to calculate the derivative of the spectral information [25].

The 1st-derivative formula is as follows:

$$f(x)' = \frac{f(x+1) - f(x)}{\Delta x}. \tag{8}$$

The 2nd-derivative formula is as follows:

$$f(x)'' = \frac{f'(x+1) - f'(x)}{\Delta x}. \tag{9}$$

3.1.2. Principal Component Analysis (PCA). The basic principle of principal component analysis (PCA) is to explore the correlation between multiple variables. The main principle of PCA is to project the original features onto the information-rich dimension to achieve dimensionality reduction, and the first few principal components (PCs) contain much of the useful information [26].

It can be assumed that the data have a total of $D = \{x_1, x_2, \cdots, x_n\}$; if you want to reduce it to the m dimension, the basic operation of PCA is as follows:

(1) Centralize all features and remove the mean value:

$$x_i \longleftarrow x_i - \frac{1}{n} \sum_{i=1}^{n} x_i. \tag{10}$$

(2) Calculate the covariance matrix XX^T / n

(3) Calculate the eigenvalues of the covariance matrix XX^T and the matching eigenvectors

(4) The original feature is projected onto the feature vector; the unit feature vector corresponding to the largest m feature values is $w_1, w_2, \cdots, w_n$; and the new m-dimensional feature after dimensionality reduction

is obtained. The output is the projection matrix $W = (w_1, w_2, \cdots, w_n)$.

3.1.3. Feature Extraction Methods.

The amount of spectral data obtained from hyperspectral images is very large and contains a lot of redundant information, which will increase the burden of data processing and affect the accuracy of the model [27]. Therefore, it is necessary to select feature wavelengths to reduce input and improve the performance of models [28].

The peaks and valleys with significant differences in the 2nd-derivative spectra can be selected as the feature wavelengths. It can eliminate interference from other backgrounds and highlight useful information in the spectra [29].

The successive projection algorithm (SPA) can be employed to compare the projection vectors of different wavelengths on others and extract the largest vector as the feature wavelength [30]. During the operation of this algorithm, a wavelength would be arbitrarily selected at first; then, the wavelength corresponding to the vector with the largest projection would be put into the variable group; it would operate in a circular selection mode until the end of the last variable.

The principle of the continuous projection algorithm is as follows. Let m be the spectra matrix of the collected samples, where n represents the number of samples, m represents the number of wavelengths, and N represents the number of variables:

(1) First, select j columns from the spectral data to form the spectral matrix x_j

(2) The remaining spectra are aggregated:

$$\Omega = \{j, 1 \le j \le J, j \notin \{k(0), \cdots, k(n-1)\}\}. \quad (11)$$

(3) Calculate the projection of the column vector:

$$P_{x_j} = x_j - \left(x_j^T x_{k(n-1)}\right) x_{k(n-1)} \left(x_{k(n-1)}^T x_{k(n-1)}\right)^{-1}, \quad j \in \Omega, \quad (12)$$

$$k(n) = \arg\left[\max\left(\left\|P_{x_j}\right\|\right)\right], \quad j \in \Omega, \text{make } x_j = P_{x_j}, j \in \Omega. \quad (13)$$

(4) Let $n = n + 1$; if $n < N$, return to equation (11), and the final feature wavelength is $\{x(k_n = 0), \cdots, N-1\}$

3.1.4. Discriminant Model.

The support vector machine (SVM) can be employed to map the raw data from a low dimension to a higher dimension and utilize hyperplanes to define the decision boundaries for classification [31]. The core of the SVM algorithm is to determine the optimal hyperplane separation class. The basic theory is illustrated as Figure 3.
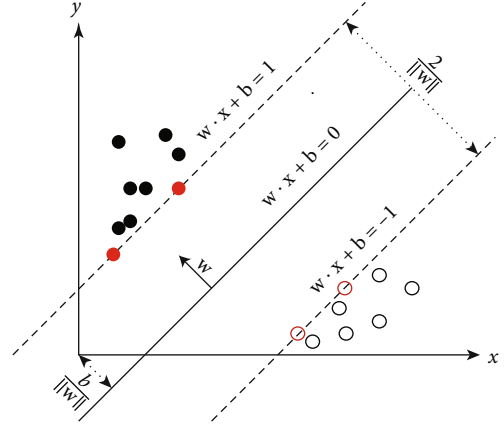
Figure 3: Schematic diagram of SVM.

This hyperplane can be described by the following equation:

$$w \cdot X_i + b = 0, \quad (14)$$

where w and b represent the normal vector and offset of the hyperplane, respectively. In the training process, the determination of the optimal hyperplane can achieve the maximum discrimination of training samples while minimizing misclassification. The two types of support vectors in the two planes parallel to the optimal hyperplane are defined as $w \cdot X_i + b = \pm 1$. Thus, the solution of this optimal plane can be transformed into a constrained optimization problem:

$$\begin{cases} \min\ \left(\frac{1}{2}\|w\|^2\right) + C\sum_{i=1}^{n} \xi_i, \\ \text{s.t.}\ \ \xi_i + y_i(w \cdot X_i + b) - 1 \ge 0, \quad \xi_i \ge 0, \end{cases} \quad (15)$$

where ξ_i represents the distance between the misclassified sample and the corresponding classification hyperplane. C represents the penalty coefficient, which determines the importance of the outlier value. According to Lagrange duality [32], the optimal hyperplane can be expressed as a combination of linear variances:

$$\begin{cases} f(x) = w \cdot X + b = \sum_{i=1}^{n} y_i \partial_i \xi_i X_i X + b, \\ \partial_i \ge 0, \end{cases} \quad (16)$$

where α_i represents the Lagrange multiplier, corresponding to the correlation coefficient of each sample, and the variation range is between 0 and C.

Kernel function is important for conducting SVM [33]. There are some popularly used kernel functions, such as polynomial kernel function, linear kernel function, sigmoid kernel function, and radial basis function (RBF). Among them, RBF has the advantages of high efficiency and fast approaching speed [34]. Simultaneously, according to the feature of small feature dimension and the normal number

Table 1: Information of samples for germination and modeling.

Accelerated aging time (days)	Germination	Modeling
0	48	48
3	48	48
6	48	48
9	48	48

of this data sample, the derivation formula can be expressed as follows:

$$K(X_{1,}X_{2,}) = \varphi(X_{1,}) \cdot \varphi(X_2) = \exp\left(-\frac{\|X_1 - X_{2,}\|}{\sigma^2}\right), \quad (17)$$

where $\varphi(X_1)$ and $\varphi(X_2)$ represent the mapping functions of objects X_1 and X_2, and σ represents the width parameters of the function.

3.1.5. Model Evaluation Index. In this study, the quality of the corresponding model was evaluated based on accuracy, root mean square error, and coefficient of determination.

(1) Accuracy. Accuracy indicates the percentage of the number of positive and negative samples that is correctly predicted to the number of all samples [35]. The overall prediction situation of the applied model can be presented. It can be expressed as follows:

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}, \quad (18)$$

where TP represents true if it is a positive sample and the model prediction is also a positive sample, TN represents true if it is a negative sample and the model prediction is also a negative sample, FP represents true if it is a negative sample but the model prediction is a positive sample, and FN represents true if it is a positive sample but the model prediction is negative sample.

(2) Root Mean Square Error (RMSE). RMSE subtracts the predicted value from the actual value, finds the square of the square and adds the average value, and finally opens the root sign [36]. The model detection ability is stronger when the RMSE is smaller. When the RMSE is 0, that is, the predicted value is completely equal to the true value, the model performance is optimal; the larger the RMSE, the greater the prediction error of the model.

The RMSE calculation formula is as follows:

$$\text{RMSE} = \sqrt{1/N \sum_n \left(y - \hat{y}\right)^2}. \quad (19)$$

(3) Coefficient of Determination (R^2). R^2 can also be called the coefficient of determination. It is mainly employed to evaluate the stability of the model based on the mean value and compared with the prediction error [37].

The R^2 calculation formula is as follows:

$$R^2 = \frac{SSR}{SST} = \frac{SSR}{SSR + SSE} = \frac{\sum\left(y_i - \hat{y}_i\right)^2}{\sum(y_i - \bar{y}_i)^2}, \quad (20)$$

where SSR represents the sum of squares due to regression and SSE represents the sum of squares due to errors.

When $R^2 = 1$, the predicted value of this model is equal to the true value of the sample, and the error is 0; when $R^2 = 0$, the predicted value of this model is equal to the mean value of the sample. R^2 can reflect the stability of the model. The more R^2 tends to 1, the more stable the model.

(4) Receiver Operating Characteristic (ROC) Curve. The ROC curve is a comprehensive index that reflects the sensitivity and specificity of continuous variables. It can be employed to calculate the sensitivity and specificity by setting different critical values and generating the ROC curve. The larger the area under the ROC curve (AUC), the better the performance of the model.

The formula for calculating the horizontal axis value is

$$\text{FPR} = \frac{FP}{TN + FP}. \quad (21)$$

The formula for calculating the vertical axis value is

$$\text{TPR} = \frac{TP}{TP + FN}, \quad (22)$$

where FPR is the false positive rate and TPR is the true positive rate.

3.2. Data Analysis Materials

3.2.1. Sample Preparation. The variety of corn seeds used in this study is a kind of waxy corn named Jingke 2000. The residual granules, overly dry granules, and impurities were removed in advance. A total of 384 seeds were selected as samples, and they were randomly divided into four groups (96 samples in each group). Information on the samples is shown in Table 1. One group was used as a control group, and the other three groups were placed in an artificial aging box for 3, 6, and 9 days of aging treatment. After the treatment was completed, 48 seeds in each group were randomly selected for standard germination analysis, and the hyperspectral data of each seed of the other half of the samples were collected every 5 minutes.

3.2.2. Standard Germination Tests. Before obtaining the hyperspectral image, the selected samples were tested for germination according to the International Seed Testing Association (ISTA) standard [38]. The corn seeds were cultured for 10 days at 25°C with a relative humidity of 99%. The sprout length was manually measured, and the germination rate was calculated according to the ISTA standard.

(a)

(b)

FIGURE 4: (a) Hyperspectral imaging system; (b) corn samples in Petri dishes.

TABLE 2: Germination rate and sprout length of Jingke 2000 under different accelerated aging periods of time.

Accelerated aging time (days)	Germination rate (%)	Average sprout length (cm)
0	91.67	1.76
3	83.33	1.01
6	60.42	0.70
9	47.92	0.40

3.2.3. Hyperspectral Imaging System. The experiment was performed using a hyperspectral imaging system (Figure 4(a)) composed of a hyperspectral imager, two halogen lamps, a mobile platform, and a computer. The hyperspectral imager SOC 710-VP manufactured by Polytec, France, was used to obtain reflected light from the seeds. It could cover a spectral range of 370–1042 nm with a spectral resolution of 4.7 nm and spatial resolution of 696×520 pixels. Each of the 24 corn seeds was placed in a specific sample holder with 24 holes (Figure 4(b)). The exposure time of the imaging system was set to 3 ms, and the platform movement speed was 20 mm/s. The obtained spectra were processed by dark and white calibration initially to calibrate the system. The postdata processing and model construction were performed by the MATLAB R2018a software (The Math Works, Natick, MA, USA).

4. Results

4.1. Germination Test Analysis. Table 2 lists the germination rate and average sprout length of waxy Jingke 2000 with different aging periods of time. It can be seen that during the aging process of corn seeds from 0 to 9 days, both the germination rate and the sprout length decrease with the increase of aging time.

The overall results show that the longer the aging time, the relatively lower the seed vitality. This indicates that it is justified to prepare samples of different vitality classes by varying aging periods of time in this study. These findings provide experimental support for later seed vitality detection with SVM at different levels of aging.

4.2. Spectral Profile. The raw spectral reflectance curves of all corn samples in the spectral range of 370–1042 nm and the average spectra of the samples at four different aging times

are shown in Figures 5(a) and 5(b), respectively. The corn seeds in the same variety have similar change trends of the spectral reflectance curves under different aging processes. The spectral reflectance gradually increases in the range of 400-650 nm, with a clear absorption peak near 650 nm; then, the reflectance decreases, reaching a minimum value in the first stage near 700 nm. After that, the curve rises sharply, reaching a maximum around 750 nm; then, it falls and rises again, reaching a minimum in the second stage around 850 nm; subsequently, the reflectance continues to fall.

The average spectra of corn seeds preprocessed by S-G smoothing and 2nd derivative are shown in Figure 6. A five-point model was applied in S-G smoothing. Compared with the unpretreated spectra, it shows that after S-G smoothing, the spurs in the original curve are obviously eliminated and the noise is smoothed. Moreover, the preprocessing with the 2nd-derivative algorithm also removes some high-frequency noise and mutual interference of different components. Therefore, it can be concluded that both the S-G smoothing and the 2nd-derivative algorithm could improve smoothness and reduce noise interference. They can be used to preprocess other continuous and irregular data.

4.3. PCA. PCA was conducted on the average spectra of corn seeds, in an attempt to obtain the weight coefficients of different PCs. The first three PCs were adopted for qualitative analysis, due to the fact that they contained much of the corn seed information, with 99.98% explained variance for Jingke 2000 (95.06% for PC1, 2.83% for PC2, and 0.94% for PC3). The contribution rate of PC1, PC2, and PC3 is shown in Figure 7(a). The weight coefficients of PC1, PC2, and PC3 are shown in Figures 7(b)–7(d), respectively. Similarly, the weight coefficients of PC2 and PC3 can also be obtained. The first three PCs were selected as the input of the models for the further processing.

4.4. Classification Models Based on Full Spectra. An SVM model was constructed to detect corn seeds with different aging periods of time. Besides, the sample data of each grade were randomly divided into a training set and a testing set at a ratio of 3 : 1 (the training set contained 36 seeds, and the testing set contained 12 seeds). The RBF kernel function was applied to this model. The grid optimization method was used for parameter optimization. The penalty coefficient c was 100 and the regularization coefficient was 10.

(a)

(b)
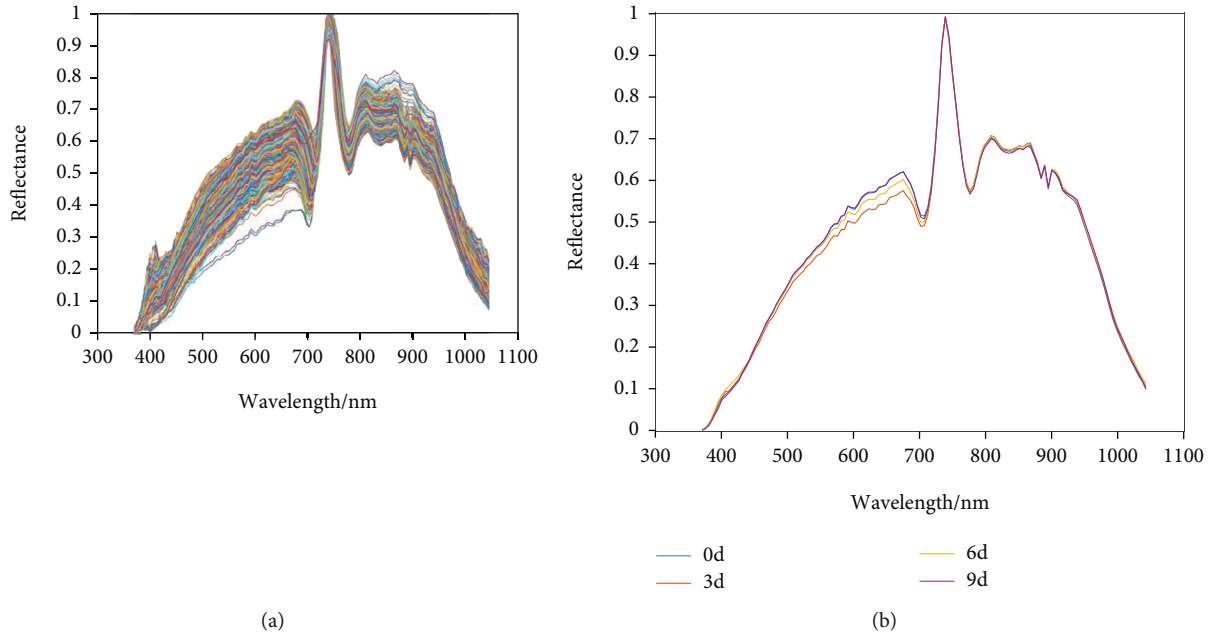
FIGURE 5: (a) Raw spectra of corn seeds; (b) average spectra of corn seeds.

(a)

(b)
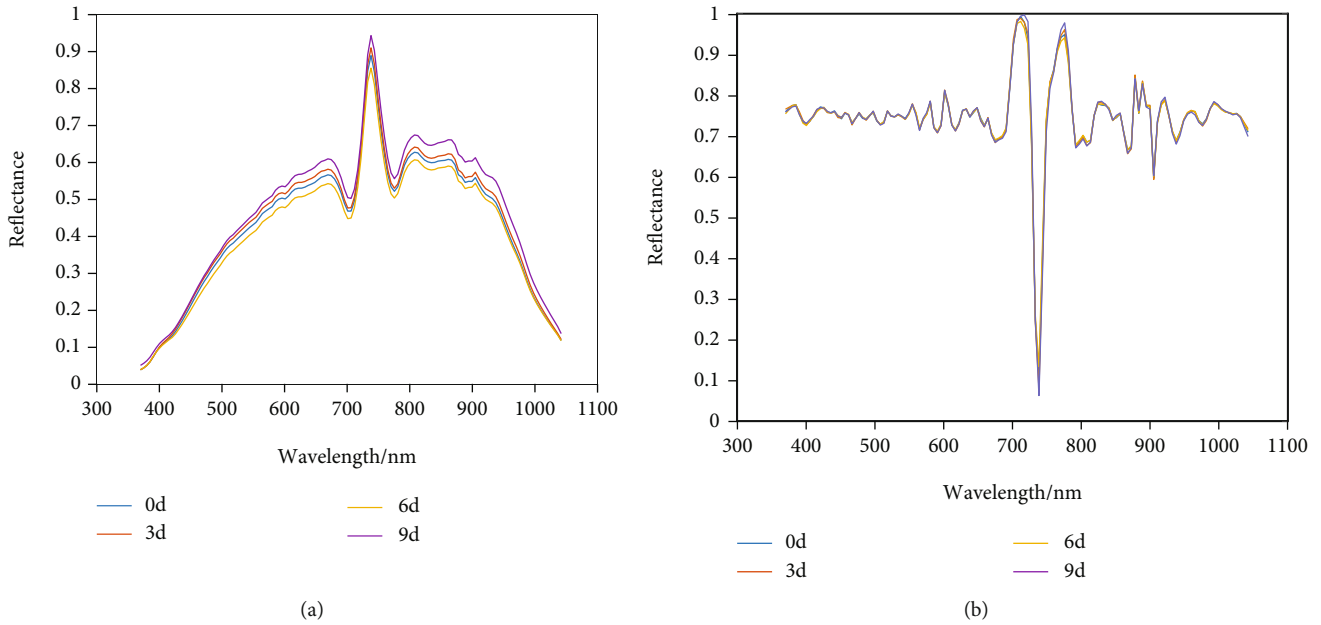
FIGURE 6: (a) Spectra preprocessed by S-G smoothing; (b) spectra preprocessed by 2nd derivative.

The overall classification results are shown in Table 3. As shown in Table 3, the overall classification accuracy of the training set for the waxy corn under PCA is relatively higher than that for the raw data. The training accuracy is 93.75% and the testing accuracy is 89.5833%. It can be inferred that PCA can reduce the dimension of the data and increase the accuracy of the classification model. As for the PCA-SVM model, RMSE is 0.0262 and R^2 is 0.834, which improves the stability of model detection errors. Due to the fact that the spectral data processed by PCA can represent as many data features as possible without relying on too many com-ponents, the calculation time of the later model is signifi-cantly reduced, and the redundant information in the raw data is removed at the same time.

4.5. Feature Wavelength Selection. In this study, the 2nd derivative and SPA were adopted to select feature wave-lengths. The RMSE of different wavelengths selected after SPA preprocessing and the selected variables are shown in Figure 8. The number of wavelengths with a locally lowest RMSE value is regarded as the number of feature wave-lengths. The selected feature wavelengths are shown in

(a)

(b)

(c)

(d)
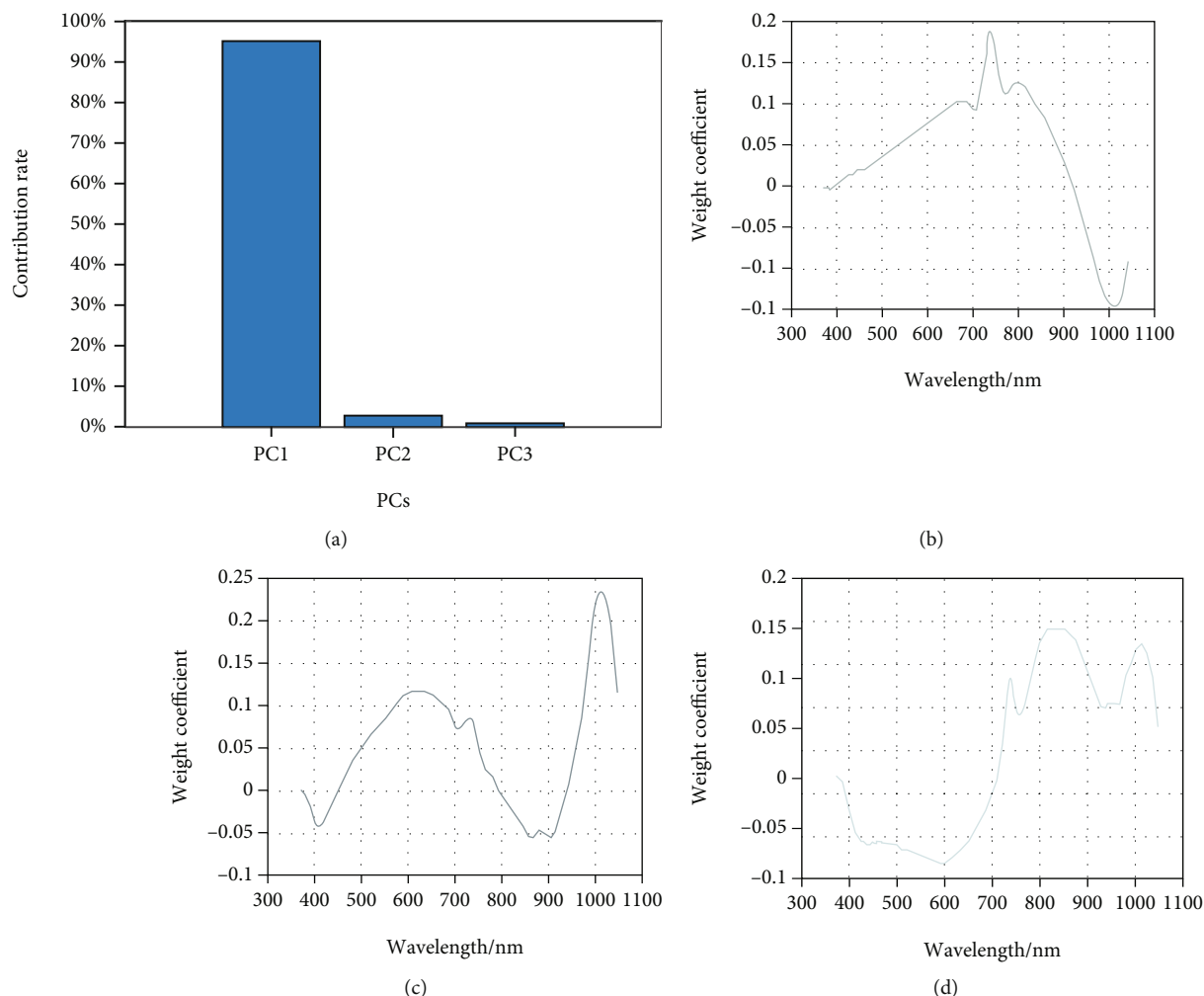
FIGURE 7: (a) Contribution rate of PC1, PC2, and PC3; (b) weight coefficient of PC1; (c) weight coefficient of PC2; (d) weight coefficient of PC3.

TABLE 3: The classification accuracy of SVM models using full spectra.

Methods	Variables	Training accuracy (%)	Testing accuracy (%)	RMSE	R^2
Raw-SVM	128	87.5	83.3333	0.05	0.834
PCA-SVM	3	93.75	89.5833	0.0262	0.9332

Table 4. A total of 20 and 8 feature wavelengths are obtained, finally, to reduce the data volume.

4.6. Classification Models Based on Feature Wavelengths.

Table 5 shows the detection results of the SVM model constructed based on the feature wavelengths selected by the 2nd derivative and SPA. The classification accuracy of both the training set and the testing set is over 95%, which is significantly higher than the overall results based on the full spectra. The highest accuracy of the training set is 98.75%, and that of the testing set is 97.1111%. As for the SPA-SVM model, RMSE is 0.0238 and R^2 is 0.9435. Through the feature wavelength selection, the effective information

is highlighted, so that the accuracy of the model is improved. Therefore, it is of great significance to adopt a classification model based on feature wavelengths. Meanwhile, SPA is more suitable for feature wavelength selection.

Figure 9 shows the ROC curves of the two classification models. The AUC of the SPA-SVM model is higher, reaching 0.9783, while the AUC of the 2nd-derivative-SVM model is 0.9416. Both the classification accuracy and the AUC of the SPA-SVM model are higher than those of the 2nd-derivative-SVM model, which indicate that the SPA-SVM model has better performance and generalization ability in the classification of corn seeds.

The overall results of SVM models using the feature wavelengths selected by SPA spectra with different processing methods are listed in Table 6.

The accuracy of the training set is higher than that of the model based on the raw spectra, all over 90%. The accuracy of the testing set is over 85%, which indicates that the SVM classification model constructed based on hyperspectral data can effectively achieve the detection of corn seeds with different vitality levels. The results of the MSC method are better than those of other pretreatment methods. The accuracy

Final number of selected variables: 128
(RMSE = 0.0579)

(a)

(b)
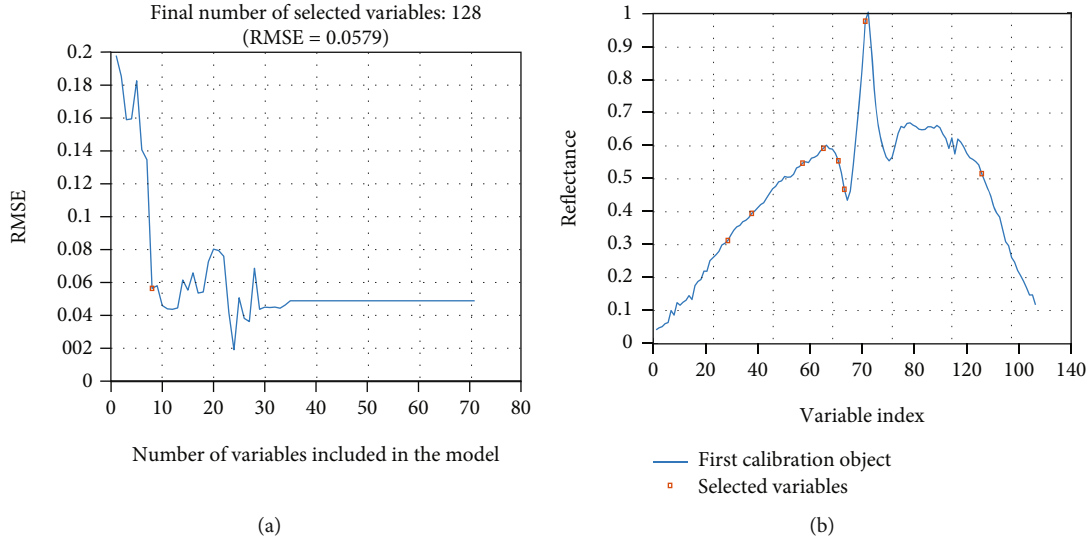
FIGURE 8: (a) RMSE distribution diagram; (b) feature wavelengths by SPA.

TABLE 4: Corresponding feature wavelengths selected by the 2nd derivative and SPA.

Methods	Variables	Feature wavelengths (nm)
2nd derivative	20	466.8984, 518.3363, 554.5661, 564.9512, 580.5571, 601.4174, 648.5726, 664.3586, 711.9191, 738.4728, 775.8057, 791.8618, 856.4238, 878.0646, 888.9074, 894.3345, 905.1999, 937.8861, 976.1908, 1014.679
SPA	8	502.8656, 544.196, 622.3378, 674.9013, 685.459, 696.0318, 738.4728, 943.3469

TABLE 5: The classification accuracy of SVM models using the 2nd derivative and SPA using feature wavelengths.

Methods	Variables	Training accuracy (%)	Testing accuracy (%)	RMSE	R^2
2nd-derivative-SVM	20	96.5278	95.8333	0.0385	0.9218
SPA-SVM	8	98.75	97.1111	0.0238	0.9435
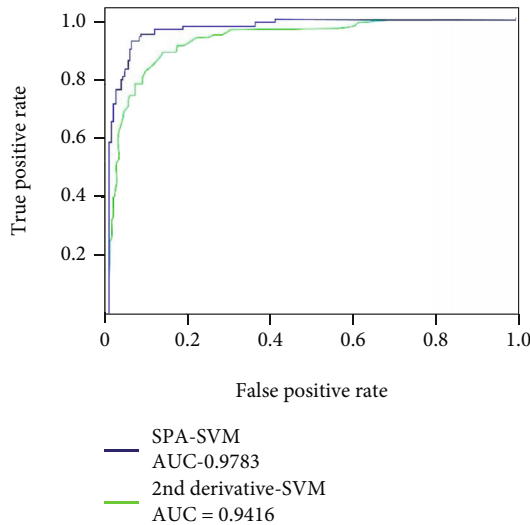
FIGURE 9: ROC curves of SPA-SVM model and the 2nd-derivative-SVM model.

of the training set is 100% and the accuracy of the testing set is 97.9167%, followed by the 2nd derivative, with the accuracy of 96.5278% in the training set and 95.8333% in the

testing set. Besides, the RMSE of the two is the lowest among all the methods, and R^2 is the highest. In terms of the MSC-SPA-SVM model, RMSE is 0.018 and R^2 is 0.875. In terms of the 2nd-derivative-SPA-SVM model, RMSE is 0.0343 and R^2 is 0.888. It can be proven that the SVM model based on feature wavelengths is better after being preprocessed with the MSC and the 2nd derivative. Further, the classification accuracy is significantly improved compared with those based on the raw spectra data. Moreover, the robustness of the models has also been improved. The preprocessing method has great significance for improving the accuracy of the classification models.

5. Discussion

As mentioned above, several preprocessing and feature wavelength selecting methods are employed to construct the SVM model based on the spectral information of Jingke 2000 waxy corn. As is revealed from Table 3, the SVM model based on PCA could achieve a significantly higher accuracy in the training set and the testing set compared with the model based on the raw spectra. The spectral data processed by PCA can represent as comprehensive features as possible with fewer PCs. The redundant information in the raw data

TABLE 6: The classification accuracy of SVM models using different processing methods based on feature wavelengths.

Methods	Number of correct recognition	Training accuracy (%)	Testing accuracy (%)	RMSE	R^2
2nd-derivative-SVM	20	96.5278	95.8333	0.0385	0.9218
SNV-SVM	133	94.4444	93.75	0.0424	0.855
MSC-SVM	144	100	97.9167	0.018	0.875
S-G smoothing-SVM	138	95.8333	91.6667	0.0359	0.864
1st-derivative-SVM	126	91.4766.5	89.2501	0.05	0.834
2nd-derivative-SVM	139	96.5278	95.8333	0.0343	0.888

is removed, and the time of the subsequent model calculation is significantly reduced. From Tables 3 and 5, it can be concluded that the detection results of the model based on the feature wavelengths have been improved compared with the model based on the full wavelengths. This indicates that feature extraction can eliminate some irrelevant information in the spectral data and make the remaining features more obvious. The ROC-AUC curve of the classification model based on the 2nd derivative and SPA shows that the classifier has better generalization performance under different thresholds. The ROC curve of the SPA-SVM model is closer to the upper left corner of the coordinate and has a larger AUC, indicating a higher accuracy. SPA can be employed to compare the projection vectors of different wavelengths on the others and extract the feature wavelengths corresponding to the largest vector. It can achieve a better performance in minimizing the correlation of the wavelengths and extracting features than other methods. After extracting the feature wavelengths by SPA, different preprocessing methods are applied to the performance comparison. The results suggest that the accuracy of the model has been further improved, with MSC achieving the optimal performance, followed by the 2nd derivative. The MSC method can be employed to remove noise and stray light interference from the raw spectra. The 2nd derivative can be adopted to eliminate the noise and the baseline drift, through which the features of the spectral curve can be highlighted. Other methods can only be adopted to smooth the noise horizontally, with a poor performance. The SVM model for the classification of waxy corn seed vitality with MSC preprocessing and the SPA method to extract the feature wavelengths could achieve the optimal performance. The accuracy of the training set is 100%, the accuracy of the testing set is 97.9167%, and there are few samples with incorrect classification. RMSE is 0.018 and R^2 is 0.875, which means that the deviation is also within the acceptable range, thus ensuring the robustness of the model. This method is faster and more nondestructive than traditional and chemical methods. The SVM model has improved the accuracy and robustness compared with other ML methods, such as PLS. Therefore, the MSC-SPA-SVM model is of great significance for the classification of Jingke 2000 waxy corn seeds.

6. Conclusions

The SVM model based on hyperspectral imaging is quite effective for detecting the waxy corn seeds with different vitality levels. The detection results of the SVM model based on the feature wavelengths combined with different preprocessing methods are generally better than the model based on the full spectra. The MSC-SPA-SVM model could achieve the highest accuracy of 97.9167% in the testing set.

Through this study, it can be concluded that it is feasible to use ML algorithms to detect the vitality of waxy corn seeds. They are fast and nondestructive during the classification compared with traditional methods. Besides, they can achieve a higher accuracy, showing great potential in the future.

However, there are some limitations and threats in the results due to the restriction of the experimental environment and other conditions. The spectral imager in this study could only cover 370 nm-1042 nm, and the wavelength range should be expanded to enrich the spectral features of the seed vitality. Therefore, it is required to explore the application of SVM and other ML algorithms in crop detection. Besides, the structure and parameters to improve the classification accuracy and versatility of the model should be optimized. In the future, it is necessary to explore its potential in data modeling, predictive analytics, and deep-learning methods [39–43]. Moreover, we will conduct further application explorations for the development of other fields including crop identification and food hazard detection [44–49].

Data Availability

All data included in this study are available upon request by contacting the corresponding author.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Conceptualization was overseen by Jinghua Wang and Lei Yan; methodology was overseen by Jinghua Wang; software was handled by Jinghua Wang; validation was overseen by Jinghua Wang and Fan Wang; formal analysis was handled by Lei Yan and Fan Wang; investigation was overseen by Lei Yan and Fan Wang; resources were overseen by Lei Yan; data curation was handled by Jinghua Wang and Lei Yan; original draft preparation was handled by Jinghua Wang; review and editing were handled by Jinghua Wang and Shanshan Qi; visualization was overseen by Jinghua

References

[1] N. Kapoor, A. Arya, M. A. Siddiqui, H. Kumar, and A. Amir, "Physiological and biochemical changes during seed deterioration in aged seeds of rice (Oryza sativa L.)," *Plant Physiology*, vol. 6, pp. 28–35, 2011.

[2] C. M. Mcdonough, C. D. Floyd, R. D. Waniska, and L. W. Rooney, "Effect of accelerated aging on maize, sorghum, and sorghum meal," *Journal of Cereal Science*, vol. 39, no. 3, pp. 351–361, 2004.

[3] S. H. Cheyed, "Effect of storage method and period on vitality and vigour of seed wheat," *Indian Journal of Ecology*, vol. 47, no. 10, pp. 27–31, 2020.

[4] J. Chen, W. Miao, K. Fei et al., "Jasmonates alleviate the harm of high-temperature stress during anthesis to stigma vitality of photothermosensitive genetic male sterile rice lines," *Plant Science*, vol. 12, 2021.

[5] C. C. Zhu, C. X. Wang, C. Y. Lu et al., "Genome-wide identification and expression analysis of OsbZIP09 target genes in rice reveal its mechanism of controlling seed germination," *International Journal of Molecular Sciences*, vol. 22, no. 4, p. 1661, 2021.

[6] J. Wei, H. Zhao, X. Liu, S. Liu, L. Li, and H. Ma, "Physiological and biochemical characteristics of two soybean cultivars with different seed vigor during seed physiological maturity," *Proteomics*, vol. 18, no. 1, pp. 71–80, 2021.

[7] M. Huang, J. B. Tang, Q. Z. Yang, and Q. Zhu, "Classification of maize seeds of different years based on hyperspectral imaging and model updating," *Computers and Electronics in Agriculture*, vol. 122, pp. 139–145, 2016.

[8] H. L. Huang, M. O. N. Liu, and M. Ngadi, "Recent developments in hyperspectral imaging for assessment of food quality and safety," *Sensors*, vol. 14, no. 4, pp. 7248–7276, 2014.

[9] B. Insuck, K. Dewi, M. K. Lalit et al., "Rapid measurement of soybean seed viability using kernel-based multispectral image analysis," *Sensors*, vol. 19, no. 2, p. 271, 2019.

[10] A. Şentaş, İ. Tashiev, and F. Küçükayvaz, "Performance evaluation of support vector machine and convolutional neural network algorithms in real-time vehicle type classification," in *International Conference on Emerging Internetworking*, Springer, Cham, 2018.

[11] L. Zhang, J. Sun, X. Zhou, A. Nirere, X. Wu, and R. Dai, "Classification detection of saccharin jujube based on hyperspectral imaging technology," *Journal of Food Processing and Preservation*, vol. 44, no. 8, 2020.

[12] M. Y. Najafabadi, H. J. Earl, and T. Dan, "Application of machine learning algorithms in plant breeding: predicting yield from hyperspectral reflectance in soybean," *Frontiers in Plant Science*, vol. 11, p. 2169, 2020.

[13] W. Cai, B. Liu, Z. Wei, M. Li, and J. Kan, "TARDB-Net: triple-attention guided residual dense and BiLSTM networks for hyperspectral image classification," *Multimedia Tools and Applications*, vol. 80, no. 7, pp. 11291–11312, 2021.

[14] Y. Y. Zheng, J. L. Kong, X. B. Jin, X. Y. Wang, T. L. Su, and M. Zuo, "CropDeep: the crop vision dataset for deep-learning-based classification and detection in precision agriculture," *Sensors*, vol. 19, no. 5, p. 1058, 2019.

[15] A. M. Fernando, A. C. Karen, S. Christy, G. Amanda, L. Renfu, and D. K. James, "Testing of canned black bean texture (Phaseolus vulgaris L.) from intact dry seeds using visible/near infrared spectroscopy and hyperspectral imaging data," *Journal of the Science of Food and Agriculture*, vol. 98, pp. 283–290, 2018.

[16] L. Wang, H. Sun, D. W. Pu, D. Liu, Q. Wang, and Z. Xiong, "Application of hyperspectral imaging for prediction of textural properties of maize seeds with different storage periods," *Food Analytical Methods*, vol. 8, no. 6, pp. 1535–1545, 2015.

[17] J. Yang, L. Sun, W. Xing, G. Feng, H. Bai, and J. Wang, "Hyperspectral prediction of sugarbeet seed germination based on gauss kernel SVM," *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, vol. 253, p. 119585, 2021.

[18] A. Medeiros, L. Silva, J. Ribeiro et al., "Machine learning for seed quality classification: an advanced approach using merger data from FT-NIR spectroscopy and X-ray imaging," *Sensors*, vol. 20, no. 15, p. 4319, 2020.

[19] M. Puneet, N. Alison, T. Julius, L. Guoping, R. Sally, and M. Stephen, "Near-infrared hyperspectral imaging for non-destructive classification of commercial tea products," *Journal of Food Engineering*, vol. 238, pp. 70–77, 2018.

[20] I. Baek, M. Kim, B. K. Cho et al., "Selection of optimal hyperspectral wavebands for detection of discolored, diseased rice seeds," *Applied Sciences*, vol. 9, no. 5, p. 1027, 2019.

[21] J. Dumont, T. Hirvonen, V. Heikkinen et al., "Thermal and hyperspectral imaging for Norway spruce (Picea abies) seeds screening," *Computers & Electronics in Agriculture*, vol. 116, pp. 118–124, 2015.

[22] T. L. Liu, Q. Su, Q. Sun, and L. M. Yang, "Recognition of corn seeds based on pattern recognition and near infrared spectroscopy technology," *Spectroscopy & Spectral Analysis*, vol. 32, no. 5, pp. 1209–1212, 2012.

[23] R. Candolfi, D. Maesschalck, R. Jouan, and D. L. Hailey, "The influence of data pre-processing in the pattern recognition of excipients near-infrared spectra," *Journal of Pharmaceutical and Biomedical Analysis*, vol. 21, no. 1, pp. 115–132, 1999.

[24] R. Sonobe, H. Yamashita, H. Mihara, A. Morita, and T. Ikka, "Estimation of leaf chlorophyll a, b and carotenoid contents and their ratios using hyperspectral reflectance," *Remote Sensing*, vol. 12, no. 19, p. 3265, 2020.

[25] H. Chen, Q. Song, G. Tang, Q. Feng, and L. Lin, "The combined optimization of Savitzky-Golay smoothing and multiplicative scatter correction for FT-NIR PLS models," *Spectroscopy*, vol. 2013, pp. 1–9, 2013.

[26] R. Jahani, H. Yazdanpanah, and S. Ruth, "Novel application of near-infrared spectroscopy and chemometrics approach for detection of lime juice adulteration," *Iranian journal of pharmaceutical research (IJPR)*, vol. 19, no. 2, pp. 34–44, 2020.

[27] S. Zhang, D. Zhang, and H. Jie, "NIR spectroscopy identification of persimmon varieties based on pca-svm," in *In*

International Conference on Computer and Computing Technologies in Agriculture, vol. 345, pp. 118–123, Springer, Berlin, Heidelberg, 2010 Oct 22.

[28] D. Liu, J. Ma, D. W. Sun et al., "Prediction of color and pH of salted porcine meats using visible and near-infrared hyperspectral imaging," *Food and Bioprocess Technology*, vol. 7, no. 11, pp. 3100–3108, 2014.

[29] Y. Zhao, S. S. Zhu, C. Zhang, X. P. Feng, L. Feng, and Y. He, "Application of hyperspectral imaging and chemometrics for variety classification of maize seeds," *Royal Society of Chemistry*, vol. 8, pp. 1337–1345, 2018.

[30] M. C. U. Araújo, T. C. B. Saldanha, R. K. H. Galvão et al., "The successive projections algorithm for variable selection in spectroscopic multicomponent analysis," *Chemometrics and Intelligent Laboratory Systems*, vol. 57, no. 2, pp. 65–73, 2001.

[31] A. Savitzky and M. J. Golay, "Smoothing and differentiation of data by simplified least squares procedures," *Analytical Chemistry*, vol. 36, no. 8, pp. 1627–1639, 1964.

[32] C. Zhang, X. Feng, J. Wang, F. Liu, Y. He, and W. Zhou, "Mid-infrared spectroscopy combined with chemometrics to detect Sclerotinia stem rot on oilseed rape (Brassica napus L.) leaves," *Plant Methods*, vol. 13, no. 1, p. 39, 2017.

[33] L. V. Utkin, "An imprecise extension of SVM-based machine learning models," *Neurocomputing*, vol. 331, pp. 18–32, 2019.

[34] O. Devos, C. Ruckebusch, A. Durand, L. Duponchel, and J. P. Huvenne, "Support vector machines (SVM) in near infrared (NIR) spectroscopy: focus on parameters optimization and model interpretation," *Chemometrics and Intelligent Laboratory Systems*, vol. 96, no. 1, pp. 27–33, 2009.

[35] P. Z. Rodrigo, T. T. Miguel, and C. F. Auat, "A pattern recognition strategy for visual grape bunch detection in vineyards," *Computers and Electronics in Agriculture*, vol. 151, pp. 136–149, 2018.

[36] A. Jiménez-Cordero, J. M. Morales, and S. Pineda, "A novel embedded min-max approach for feature selection in nonlinear support vector machine classification," *European Journal of Operational Research*, vol. 293, no. 1, pp. 24–35, 2021.

[37] J. Omar, A. Boix, and F. Ulberth, "Raman spectroscopy for quality control and detection of substandard painkillers," *Vibrational Spectroscopy*, vol. 111, article 103147, 2020.

[38] S. Singha, S. Pasupuleti, S. S. Singha, and S. Kumar, "Effectiveness of groundwater heavy metal pollution indices studies by deep-learning," *Journal of Contaminant Hydrology*, vol. 235, no. 6, article 103718, 2020.

[39] J. Kong, H. Wang, X. Wang, X. Jin, X. Fang, and S. Lin, "Multistream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture," *Computers and Electronics in Agriculture*, vol. 185, article 106134, 2021.

[40] J. Kong, C. Yang, J. Wang et al., "Deep-stacking network approach by multisource data mining for hazardous risk identification in IoT-based intelligent food management systems," *Computational Intelligence and Neuroscience*, vol. 2021, 1194516 pages, 2021.

[41] J. Xuebo, Z. Weizhen, K. Jianlei et al., "Deep-learning forecasting method for electric power load via attention-based encoder-decoder with Bayesian optimization," *Energies*, vol. 14, no. 6, p. 1596, 2021.

[42] X. B. Jin, W. Z. Zheng, J. L. Kong et al., "Deep-learning temporal predictor via bi-directional self-attentive encoder-decoder framework for IOT-based environmental sensing in intelligent greenhouse," *Ariculture*, vol. 11, no. 8, p. 802, 2021.

[43] X.-B. Jin, W.-T. Gong, J.-L. Kong, Y.-T. Bai, and S. Ting-Li, "PFVAE: a planar flow-based variational auto-encoder prediction model for time series data," *Mathematics*, vol. 10, no. 4, p. 610, 2022.

[44] X.-B. Jin, W.-T. Gong, J.-L. Kong, Y.-T. Bai, and S. Ting-Li, "A variational Bayesian deep network with data self-screening layer for massive time-series data forecasting," *Entropy*, vol. 24, no. 3, p. 335, 2022.

[45] X.-B. Jin, J.-S. Zhang, J.-L. Kong, S. Ting-Li, and Y.-T. Bai, "A reversible automatic selection normalization (RASN) deep network for predicting in the smart agriculture system," *Agronomy*, vol. 2, p. 1587277, 2022.

[46] J.-L. Kong, H.-X. Wang, C.-C. Yang, X.-B. Jin, M. Zuo, and X. Zhang, "A spatial feature-enhanced attention neural network with high-order pooling representation for application in pest and disease recognition," *Agriculture*, vol. 12, 2022.

[47] Y. Tong, L. Yu, S. Li, J. Liu, H. Qin, and W. Li, "Polynomial fitting algorithm based on neural network," *ASP Transactions on Pattern Recognition and Intelligent Systems*, vol. 1, no. 1, pp. 32–39.

[48] X. Ning, Y. Wang, W. Tian, L. Liu, and W. Cai, "A biomimetic covering learning method based on principle of homology continuity," *ASP Transactions on Pattern Recognition and Intelligent Systems*, vol. 1, no. 1, pp. 9–16, 2021.

[49] L. Zhang, X. Wang, X. Dong, L. Sun, W. Cai, and X. Ning, "Finger vein image enhancement based on guided tri-Gaussian filters," *ASP Transactions on Pattern Recognition and Intelligent Systems*, vol. 1, no. 1, pp. 17–23, 2021.

Research Article

Improved TLBO for Fusion of Infrared and Visible Images

Jinghua Wang [iD],[1,2] **Lei Yan** [iD],[1] **Fan Wang,**[1] **and Shulin Li** [iD][3]

[1]*School of Technology, Beijing Forestry University, Key Lab of State Forestry Administration for Forestry Equipment and Automation, Beijing 100086, China*
[2]*National Engineering Laboratory for Agri-Product Quality Traceability, Beijing 100048, China*
[3]*School of Health Sciences, The University of Manchester, Oxford Rd, Manchester, M13 9PL, UK*

Correspondence should be addressed to Lei Yan; mark_yanlei@bjfu.edu.cn

Image fusion is an image enhancement method in modern artificial intelligence theory, which can reduce the pressure in data storage and obtain better image information. Due to different imaging principles, information of the infrared image and visible images' information is complementary and redundant. The infrared image can be fused with a visible image to obtain both the high-resolution texture details and the edge contour of the infrared image. In this paper, the fusion algorithm of forest sample image is studied at the feature level, which aims to accurately extract tree features through information fusion, ensure data stability and reliability, and improve the accuracy of target recognition. The main research contents of this paper are as follows: (1) teaching learning-based optimization (TLBO) algorithm was used to optimize the weighted coefficient in the fusion process, and the value range of random parameters in the model was adjusted to optimize the fusion effect. Compared with before optimization, image information increased by 2.05%, and spatial activity increased by 15.27%. (2) Experimental data show that the target recognition accuracy of feature-level fusion results was 93.6%, 13.9% higher than that of the original infrared sample image, and 18.8% higher than that of the original visible sample image. Pixel-level and feature-level fusion have their characteristics and application scopes. This method can improve the quality of the specified region in the image and is suitable for detecting intelligent information in forest regions.

1. Introduction

With the rapid development of sensor technology, single visible light mode is gradually developed into a variety of sensor modes. They differ in imagining mechanism, working environment, and requirements as well as functions. They also work in different wavelength ranges. Due to the limited information of data acquired by a single sensor, it is often difficult to meet the needs of applications. At the same time, more comprehensive and reliable information of observation targets can be obtained by using multisource data. Therefore, in order to take full advantage of increasingly complex source data, various data fusion techniques have been rapidly developed with the aim of incorporating more supplementary information into a new data set by means of more information than can be obtained from any single

sensor [1]. Image fusion technology, as a very important branch of multisensor and visual information fusion, has aroused widespread concern and research upsurge in the world in the past twenty years. The main idea of image fusion is to combine multisource images from multiple sensors into a new image by using algorithms, so that the fused image has higher reliability, less uncertainty, and better comprehensibility [2].

Image fusion technology was first used in remote sensing image analysis and processing. In 1979, Daily et al. first applied the composite image of radar image and Landsat-MSS image to geological interpretation, and its processing process can be regarded as the simplest image fusion [3]. In 1981, Laner and Todd conducted a fusion experiment of Landsat-RBV and MSS image information [4]. In the middle and late 1980s, image fusion technology has been applied to

the analysis and processing of remote sensing multispectral images, beginning to attract attention. It was not until the end of 1980 that people began to apply image fusion technology to general image processing (visible image, infrared image, etc.) [5]. Since the 1990s, the research of image fusion technology has been on the rise, showing great application potential in the fields of automatic recognition, computer vision, remote sensing, robotics, medical image processing, and military applications. For example, the fusion of infrared and low-light images helps soldiers see targets in the dark [6]. The fusion of CT and MRI images is helpful for doctors to diagnose diseases accurately [7]. Jin et al. extracted more accurate and reliable feature information from images by fusion of infrared and visible images, thus achieving accurate face recognition [8]. Using image fusion, Liu et al. made images with different focal lengths complement each other and improve the resolution of fusion results [9]. In recent years, image fusion has become an important and useful technique for image analysis and computer vision.

The main purpose of this paper is to find an image fusion algorithm suitable for forest environment perception, using visible light image and infrared thermal image fusion technology, to collect the image fusion processing, improve the fusion effect, accurately extract effective forest information, and obtain information for forest intelligent detection. The main research contents are as follows:

(1) The fusion background of visible and infrared images, different image processing methods, and the effects of different image fusion processing are introduced

(2) The process of fusion coefficient optimization based on teaching learning based optimization (TLBO) algorithm is introduced. The random parameters in the model are set by TLBO optimization algorithm to optimize the fusion effect. The forest images are used for image fusion experiments, and the fusion results are evaluated by objective evaluation indexes

(3) In order to enhance the search ability of the algorithm and improve the evaluation index value to a greater extent, the value range of the optimization coefficient Ri and T_f of TLBO algorithm is further set according to the entropy value, and then evaluation index is used for corresponding evaluation

2. Related Works

Multisource image fusion algorithm also has broad application prospects in the field of forestry intelligent detection. Using feature-level image fusion algorithm, Bulanona et al. extracted data information of fruits in fruit forests and monitored fruit growth status in real time in 2009 [10]. In 2013, Lei et al. identified obstacles in forest images by using the results obtained by fusion algorithm and two-dimensional laser data and intelligently and accurately distinguished trees, rocks, and animals in the images with an accuracy rate of more than 93.3% [11]. Furthermore, by improving the

fusion algorithm, the data accuracy of objects such as trees in the image is improved, and the accuracy of target recognition is increased by 95.3% [12]. The quality of information fusion directly affects the accuracy of forest information detection and is an important part of research on artificial intelligence. This paper is an important branch of research on information fusion algorithm-infrared and visible image fusion algorithm. Due to different imaging principles, the information of infrared image and visible image is complementary and redundant. The target in infrared image has clear edge features and is easy to be segmented and extracted. The texture details and background information of visible image are more prominent, but the target information is difficult to extract because of complex image content. Therefore, the purpose of fusion is to synthesize complementary information, reduce redundancy, improve image quality, and express and extract useful features in images more succinctly and accurately. In this paper, the effective forest information is extracted accurately by fusion of infrared and visible images, and the obtained information is used for intelligent forest detection.

3. Materials and Methods

The detailed process of visible and near-infrared image sample fusion is shown in Figure 1. The ultimate goal is to enhance the search ability of the algorithm, improve the evaluation index value to a greater extent, and obtain the fusion image more suitable for the intelligent detection of forest information.

3.1. Data Modeling Methods

3.1.1. Pixel Level Image Fusion Algorithm

(1) Image Fusion Algorithm Based on Wavelet Transform. Wavelet transform theory was first proposed by Morlet and Gorsmsna in 1984, and its principle is developed on the basis of Fourier transform. Different from Fourier transform, wavelet transform is the local transform of frequency, which can effectively extract the signal in the image. Its advantage is to carry out multiscale analysis of the image without losing the information [13]. Stephane and Matllat proposed fast discrete wavelet and built a bridge between wavelet transform and multiscale image fusion [14].

Two-dimensional image samples after wavelet decomposition can be represented by four subband components:

$$f(x, y) = A_j f + D_j^1 f + Dj_1^2 f + D_j^3 f, \tag{1}$$

$$A_j f = \left\langle f(x, y), \varnothing_{j,m,n}(x, y) \right\rangle, \tag{2}$$

$$D_j^1 f = \left\langle f(x, y), \psi_{j,m,n}^1(x, y) \right\rangle, \tag{3}$$

$$D_j^2 f = \left\langle f(x, y), \psi_{j,m,n}^2(x, y) \right\rangle, \tag{4}$$

$$D_j^3 f = \left\langle f(x, y), \psi_{j,m,n}^3(x, y) \right\rangle, \tag{5}$$
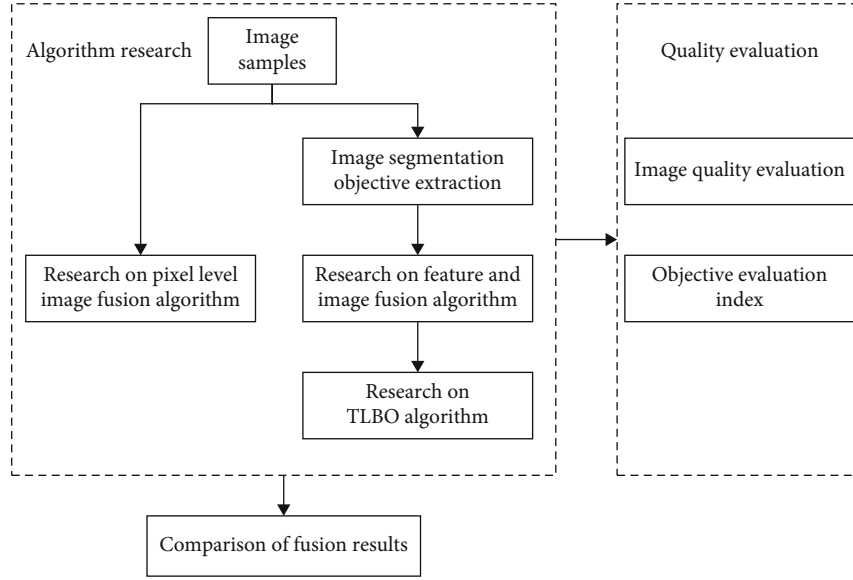
FIGURE 1: Flowchart for research on infrared and visible image fusion algorithm.

where f represents the original sample, j represents the decomposition frequency of this layer, A represents the low-frequency component, and D represents the high-frequency component in different directions. $\varnothing_{j,m,n}$ is the scale coefficient that makes up the canonical orthogonal basis of the wavelet, and the wavelet function $\psi_{j,m,n}$ makes up the canonical orthogonal basis of the space.

(2) Image Fusion Algorithm Based on PCA Transform. Principal component analysis (PCA) transform, also known as principal component analysis, is a multidimensional linear transform based on the statistical characteristics of images, which has the function of centralizing variance information and compressing data volume is mathematically called $K - L$ transform.

The PCA transformation and fusion process of multisensor images is as follows:

(1) PCA was applied to the low-resolution multispectral image to obtain three principal components: P1, P2, and P3

(2) The high-resolution image was stretched and made to have the same mean and variance as the first principal component P1 of the multispectral image

(3) The stretched high-resolution image was used to replace P1 as the first principal component, and a new fusion image P was generated with components P2 and P3 through PCA inverse transformation

(3) Image Fusion Algorithm Based on Contourlet Transform. Contourlet transform, also called contourlet transform, is a multiresolution image representation method proposed by

Do and Vetterli in 2002 [15]. In contourlet transform, image multiscale decomposition is realized by Laplace tower decomposition (LP) [16]. The multiscale decomposition of an approximate image can be obtained by repeated Laplacian tower decomposition [17]. However, in the process of image decomposition and reconstruction by contourlet transform, the image needs to be further sampled and upward sampled, which makes the contourlet transform lack shift-invariance (invariance) [18]. As a result, the spectrum of the signal will overlap to some extent, and the Gibbs phenomenon is obvious in the image fusion.

(4) Low-Frequency Coefficient Processing Based on PCNN. In the low-frequency domain of image fusion, Laplacian energy, as excitation input to PCNN, is processed as follows:

$$\begin{aligned} \mathrm{ML}(i,j) = {}& |2I(i,j) - I(i-\text{step},j) - I(i+\text{step},j)| \\ & + |2I(i,j) - I(i,j-\text{step}) - I(i,j+\text{step})| \\ & + |2I(i,j) - I(i-\text{step},j-\text{step}) - I(i+\text{step},j+\text{step})| \\ & + |2I(i,j) - I(i-\text{step},j+\text{step}) - I(i+\text{step},j+\text{step})|, \end{aligned} \tag{6}$$

where step represents the variable distance between the coefficients (in this paper, step = 1); $I(i,j)$ is the coefficient at point (i,j).

In order to eliminate the block effect or grayscale distortion that may be caused by the boundary discontinuity at the junction between the clear area and the fuzzy area, the sum of modified Laplacian (SML) in the field centered on point (i,j) is defined as

$$\mathrm{SML}(i,j) = \sum_{p}\sum_{q} W(p,q)[\mathrm{ML}(i+p,j+q)]^2, \tag{7}$$
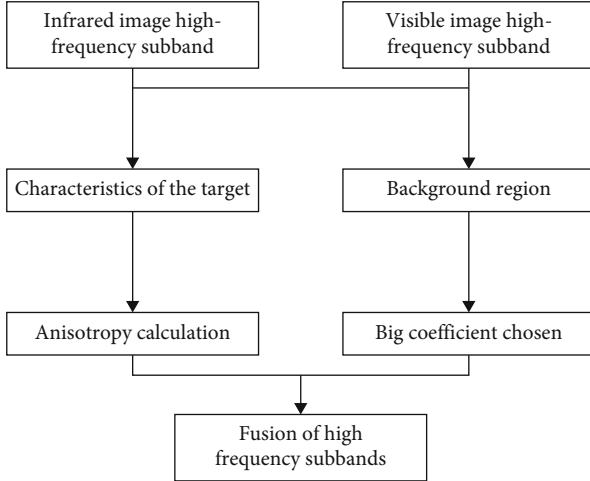
FIGURE 2: Flowchart for high-frequency fusion.

where $W(p, q)$ is the corresponding window function. Experience shows that the best highlighting effect of the window center pixel and its changing boundary should be set as

$$W(p, q) = \frac{1}{15} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 1 \end{bmatrix}. \tag{8}$$

The sum of modified Laplacian can well represent the edge details of the image, reflect the sharpness of the image, and show superior fusion performance in the fusion image.

(5) High-Frequency Coefficient Processing Based on PCNN. The high-frequency subband image represents the edge details of the image, so the coefficients decomposed by NSCT can be directly input into PCNN as excitation in the process of high-frequency coefficient fusion. The specific steps are as follows:

The PCA transformation and fusion process of multisensor images is as follows:

(1) In high-frequency subband images, the normalized gray values of each pixel are directly taken as the external input of PCNN to calculate the ignition times of each input excitation. The formula is expressed as

$$T_{ij}^k(n) = T_{ij}^k(n-1) Y_{ij}^k(n). \tag{9}$$

(2) The processing steps for the same low-frequency coefficient are as follows:

$$D_{ij,F}^k(N_1) = \begin{cases} 1, & \text{if} : T_{ij,A}^k(N_1) \geq T_{ij,B}^k(N_1), \\ 0, & \text{others}, \end{cases} \tag{10}$$

$$I_F^k(i, j) = \begin{cases} I_A^k(i, j), & \text{if} : D_{ij,F}^k(N_1) = 1, \\ I_B^k(i, j), & \text{if} : D_{ij,F}^k(N_1) = 1, \end{cases} \tag{11}$$

where $I_F^k(i, j)$, $I_A^k(i, j)$, and $I_B^k(i, j)$, respectively, represent the gray values of the fusion image and the original image A and $B(i, j)$, and k represents the NSCT decomposition of the k-layer. After each fusion subband image is obtained, the fusion image is obtained by NSCT inverse transformation.

3.1.2. Feature-Level Fusion Algorithm

(1) Low-Frequency Domain Fusion Rule Based on Fuzzy Logic. Based on fuzzy rules, fusion can also be divided into two types: spatial domain fusion and frequency domain fusion. Teng et al. fuzzified all pixel points into five fuzzy subsets based on the gray value of the image, then determined the membership degree of each fuzzy subset in the corresponding domain by a triangular membership function, and formulated fusion rules on this basis to obtain fusion results [19]. Cai and Wei first decomposed the source image into the frequency domain and then formulated the fusion rules in the low-frequency domain by using the fuzzy logic criterion to maximize the information content of the fusion sub-band image in the low-frequency domain [20]. In this paper, Gaussian membership function is used to determine the weight coefficient of image fusion, whose definition is expressed as

$$\lambda_1(i, j) = \exp\left[-\frac{(f_1(i, j) - \mu)^2}{2(k\sigma)^2}\right], \tag{12}$$

where σ is the standard deviation of the sub-band image, $f_1(\mathbb{X}, j)$ is the low-frequency decomposition coefficient of point (i, j), μ is the average value of the decomposition coefficient, and k is a constant.

(2) High-Frequency Domain Fusion Rules Based on Segmentation Results. The role of high-frequency fusion rules is to solve the problem that the target is not significant in the pixel-level fusion results and then improve the texture energy and other features of trees in the fusion results, so that the tree features extracted from the fusion results are more accurate. The flow chart is shown in Figure 2.

(3) Fusion Coefficient Optimization Algorithm. Teaching-learning-based optimization algorithm (TLBO) is a swarm intelligence algorithm proposed by Rao et al., an Indian scholar, in 2011 [21]. It imitates the learning process model of students and can be divided into two parts: teaching and learning phases. In 2014, Jin and Wang first applied TLBO optimization algorithm to image fusion to optimize the fusion coefficient and improve the image quality evaluation index [22].
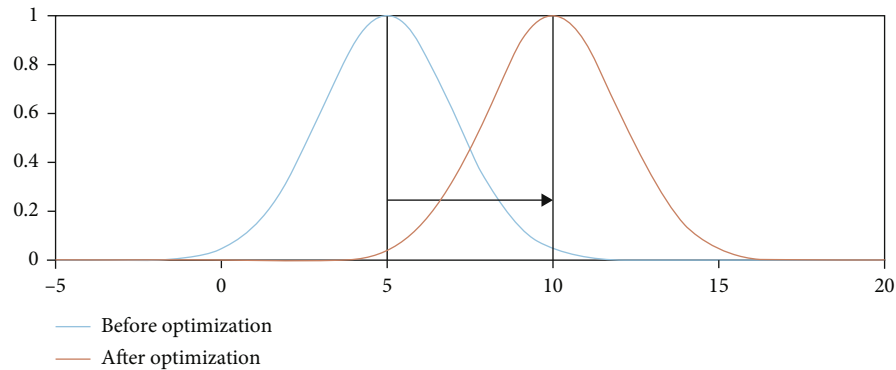
FIGURE 3: Schematic diagram of optimization process.

(1) Teaching phase

In the teaching phase, the overall optimization can be achieved by encouraging top students. Figure 3 shows the schematic diagram of the overall optimization process in this phase.

As shown in the curve for a group of student's overall academic record in Figure 3, the result agrees with the normal distribution, with its average representing students' overall level. In each optimization process, the teacher scored the best by students is defined, and then the level of the teacher is further optimized in the overall level of students through their influence.

(2) Learning phase

In the learning phase, the target function index can be improved through mutual learning among individuals. The process is carried out according to the following rules:

$$
\begin{aligned}
\text{if } f(x_i) < f(x_j), \quad & x_{\text{new},i} = x_{\text{old},i} + r_i (x_i - x_j), \\
\text{else,} \quad & x_{\text{new},i} = x_{\text{old},i} + r_j (x_j - x_i).
\end{aligned}
\tag{13}
$$

Compared with PSO, GA, and other optimization algorithms, the coefficient in TLBO has less influence on the optimization effect, with better convergence but requires shorter optimization time.

The detailed optimization steps are as follows:

(a) The weight coefficients determined by Gaussian membership function during image fusion were converted into a row of vectors, which were used as a group of samples. Another 9 groups of vectors with the same size were randomly generated to form the model to be tested

(b) The entropy value of fusion image was selected as the objective function

(c) The model was put into the TLBO system, and the fusion coefficient group under the optimal entropy value was obtained through the cycle until the convergence of objective function

(d) The cycle was terminated

(4) *Improved TLBO Parameter Optimization Algorithm.* The basic TLBO algorithm can find the global optimal value when solving simple low-dimensional problems, but when solving complex multimode high-dimensional problems, it is easy to fall into the local optimal value and cannot find the values adjacent to the global optimal value. Many scholars have improved the TLBO algorithm. Rao et al. supplemented and improved the structure of TLBO algorithm. Gao et al. introduced the crossover operation of differential evolution algorithm into the algorithm to further improve the local search ability of the algorithm [23]. All the indexes of the image optimized by using the basic TLBO algorithm were improved but not quite significantly. Therefore, in order to enhance the search ability of the algorithm and improve the evaluation index value to a greater extent, the value range of optimization coefficients R_i and T_f of TLBO algorithm was further adjusted.

The detailed optimization steps of the improved TLBO are as follows:

(1) The weight coefficients determined by Gaussian membership function during image fusion were converted into a row of vectors, which are used as a group of samples. Another 9 groups of vectors with the same size were randomly generated to form the model to be tested

(2) The entropy value of fusion image was selected as the objective function

(3) The value range of T_f was kept unchanged, the range of parameter R_i was set to compare the influence of R_i in different ranges on the image entropy, and then the optimal R_i was selected

(4) The model was put into the TLBO system, and the fusion coefficient group under the optimal entropy value was obtained through the cycle until the convergence of the objective function

(5) The R_i value range was kept unchanged, the parameter T_f range was set to compare the influence of T_f

FIGURE 4: Fluke Ti55 infrared thermal imaging camera.

in different ranges on the image entropy value, and then the optimal T_f was selected

(6) The model was then brought into the TLBO system, and the fusion coefficient group under the optimal entropy value was obtained through the cycle until the convergence of objective function

(7) The cycle was terminated

3.1.3. The Evaluation Index. In this paper, information entropy, mean gradient, standard deviation, spatial resolution, and interactive information are selected as image evaluation indicators.

(1) Information Entropy. Information entropy is the most widely used objective evaluation index of images at present, which quantitatively describes the information contained in images, and its mathematical definition is expressed as

$$E = -\sum_{i=0}^{255} P_i \log P_i, \tag{14}$$

where E represents information entropy, and P represents the proportion of the number of pixels with gray value of I in the total pixel points. The larger the information entropy is, the more scattered the gray value of image pixels is, the richer the content is, the larger the information is, and better the fusion effect is.

(2) Average Gradient. The average gradient reflects the difference between adjacent pixels in the image. The larger the average gradient is, the greater the image contrast is, the more obvious the edge effect of objects in the image is, and the clearer the texture details are. The mean gradient is defined as

$$\overline{\text{grad}} = \frac{1}{(m-1)(n-1)} \sum_{m-1} \sum_{n-1} \sqrt{\frac{(F(i,j) - F(i+1,j))^2 + (F(i,j) - F(i,j+1))^2}{2}}, \tag{15}$$

where $\overline{\text{grad}}$ represents the average gradient; m and n are the size of the image; $F(i,j)$ represents the pixel gray value of coordinate (i,j).

(3) Standard Deviation. Standard deviation represents the dispersion degree of pixel gray value distribution. The larger the value is, the more discrete the gray value distribution of image pixels is, and the stronger the contrast. The mathematical expression of standard deviation is defined as

$$\text{std} = \sqrt{\frac{\sum (F(i,j) - \bar{F})^2}{(n-1)}}, \tag{16}$$

where STD stands for standard deviation, $F(i,j)$ stands for pixel value at point (i,j), and $\bar{F}$ stands for pixel mean of all pixel points.

(4) Spatial Resolution. Spatial frequency is a parameter used to represent the activity degree of images in space. The higher the value is, the higher the activity degree of images in space is and the better the quality of images is. The formula of spatial frequency is expressed as

$$\text{RF} = \sqrt{\frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=2}^{N} [F(i,j) - F(i,j-1)]^2}, \tag{17}$$

$$\text{CF} = \sqrt{\frac{1}{M \times N} \sum_{i=2}^{M} \sum_{j=1}^{N} [F(i,j) - F(i-1,j)]^2}, \tag{18}$$

$$\text{SF} = \sqrt{\text{RF}^2 + \text{CF}^2}, \tag{19}$$

where SF is spatial frequency, RF and CF represent spatial column frequency and spatial row frequency, respectively. M, N represent the number of rows and columns of the image, and $F(i,j)$ is the gray value of pixel point (i,j).

(5) Interactive Information. Interactive information, also known as mutual information, is usually used to demonstrate the correlation between multiple variables. In the image quality evaluation system, it is used to evaluate the correlation between fusion results and original samples. The greater the amount of interaction information, the higher the correlation between the fusion result and the original sample, and the more information can be obtained from the original sample:

$$\text{MI}_{\text{AF}} = E(A) + E(F) - E(\text{AF}), \tag{20}$$

$$\text{MI}_{\text{BF}} = E(B) + E(F) - E(\text{BF}), \tag{21}$$

$$\text{MI} = \text{MI}_{\text{AF}} + \text{MI}_{\text{BF}}, \tag{22}$$

where MI is the interactive information, A, B represents the original sample, F is the fusion result, and E is the image entropy value.

3.2. Data Analysis Materials. Nearly 400 groups of forest infrared and visible images were collected in this study. The equipment used is Fluke TI55 infrared thermal imager. The time period selected in this experiment is the morning and evening when the temperature difference is large, and

TABLE 1: Technical parameters of Fluke Ti55.

	Visible lens	Infrared lens
Detector type	1280*1024 full color pixel	320*240 focal plane array
Calibration temperature range	-20~600°C	-20~600°C
Visual angle	—	23° * 17°
Spatial resolution	0.47 mrad	1.30 mrad
Minimum focus	0.6 m	0.15 m
Accuracy	2%	—
NETD	—	≤0.05°C
Spectral band	—	$8 \sim 14\,\mu$m
Detector type	1280*1024 full color pixel	320*240 focal plane Array
Calibration temperature range	-20~600°C	-20~600°C

(a)   (b)

FIGURE 5: Examples of infrared image and visible image ((a) infrared image; (b) visible image).

(a)   (b)

(c)
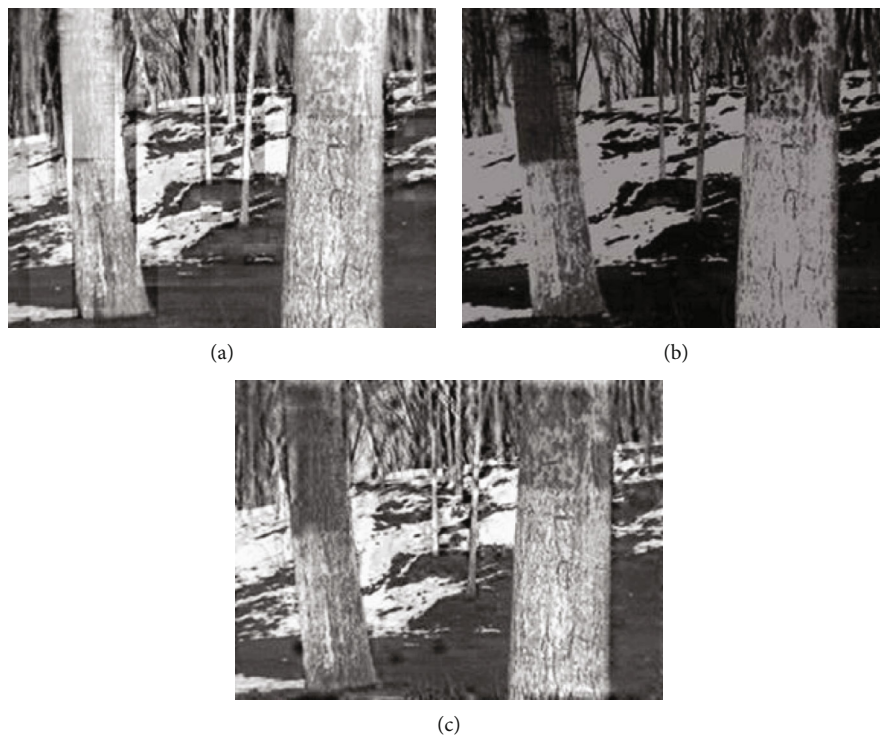
FIGURE 6: Pixel level fused result ((a) wavelet transform; (b) PCA; (c) contourlet and PCNN).

TABLE 2: Quality assessment of pixel-level fusion result.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Wavelet transform	7.5953	62.1048	19.8465	28.9314	5.4801
PCA transform	6.8579	55.7364	16.6849	23.3580	5.2299
Contourlet +PCNN	7.6367	56.3843	18.6997	29.2188	5.5820

(a)   (b)

FIGURE 7: Fusion result of pixel level and region-based level ((a) pixel-level; (b) region-based level).

TABLE 3: Quality assessment of pixel-level and region-based level fusion results.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Pixel-level image fusion results	7.6367	56.3843	18.6997	29.2188	5.5820
Feature level image fusion results	7.3981	46.2270	15.4640	29.9097	5.8198

TABLE 4: Quality assessment of region-based level and TLBO optimization fusion results based on image samples.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Region-based level	7.3981	46.2270	15.4640	29.9097	5.8198
TLBO optimization	7.5121	54.9715	17.4434	33.1374	6.0479

(a)   (b)

FIGURE 8: Fusion result of region-based level and TLBO optimization based on image samples ((a) region-based level; (b) TLBO optimization).

Table 5: Quality assessment of region-based level and TLBO optimization fusion results based on the other group images.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Region-based level	7.2378	38.5448	18.2632	31.9562	5.7235
TLBO optimization	7.3910	45.8079	18.8261	32.5719	5.2021

(a)   (b)
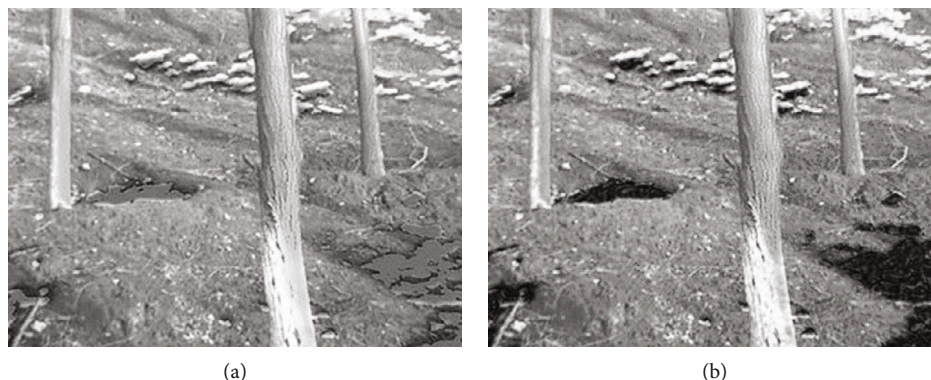
Figure 9: Fusion result of region-based level and TLBO optimization based on the other group images ((a) region-based level; (b) TLBO optimization).

the afternoon when the visual effect is easily affected. The image sample collection experiment and experimental equipment of this study are shown in Figure 4, and its technical parameters are shown in Table 1.

The infrared lens captures the spectral information in the 8-14 band, which is the middle and far infrared image. The contour of forest edge in infrared samples is obvious and thus easy to be segmented and extracted, but the accuracy of target recognition cannot be guaranteed due to the lack of details such as texture. Visible light samples contain rich texture details, but there is a very small gray difference between trees and background area without any pronounced characteristics, so it is difficult to achieve the stable extraction of tree information alone. Therefore, the purpose of this study is to improve image quality and accurately extract forest tree feature information by integrating the characteristics of infrared and visible images through fusion processing to ensure the accuracy of recognition.

4. Results

4.1. Pixel-Level Image Fusion Results. Figure 5 shows samples of infrared and visible forest images. Figure 6 shows the results of wavelet decomposition algorithm, PCA fusion algorithm, and contoulet combined with PCNN fusion algorithm, respectively. In wavelet transform, the image is decomposed into a low-frequency domain and a high-frequency domain in three directions, including horizontal high-frequency domain, vertical high-frequency domain, and oblique high-frequency domain. Wavelet decomposition can overcome the instability of Laplace decomposition and effectively reduce the influence of noise on the image. However, due to the defect of wavelet decomposition basis, jagged block error is likely to occur when processing smooth curves. PCA transformation of the principal component information is relatively high, using the gray value of pan-

chromatic band image to replace PCA, and then inverse transformation of the enhanced multispectral band image, the information is vulnerable to loss. Contourlet decomposition +PCNN transform can avoid block effect and grayscale distortion while improving image definition and contrast, avoiding generating new noise and expanding the information of a single image.

From the perspective of subjective evaluation, the fusion results synthesize the features of the source image, expand the information content of a single image, and improve the image clarity. Based on contourlet decomposition, this algorithm is a multiscale and multidirection computing framework for discrete images. It can be regarded as the enhancement technology of contourlet decomposition, which can carry out multidirection decomposition and multiscale decomposition of images, respectively. By the improved method, contourlet decomposition +PCNN transform can eliminate the aliasing effect caused by using contourlet and provide a good and stable input signal for the subsequent fusion.

In order to quantitatively evaluate the quality of the fusion results, the image was quantitatively analyzed, as shown in Table 2. As can be seen from the table, the standard deviation and mean gradient data distortion of wavelet fusion are caused by the fact that wavelet transform cannot effectively process the smooth curve in the image, which is likely to result in the jagged noise and interferes with the statistical characteristics of the image. Contourlet decomposition +PCNN transform can avoid block effect and grayscale distortion, while improving image definition and contrast, avoiding generating new noise, and expanding the information of a single image. It can also more accurately describe the forest area of the tree information and its scene details. In terms of quantitative data analysis, the results obtained by contourlet combined with PCNN algorithm are better than those obtained by the other two algorithms

(a)

(b)

(c)

(d)

(e)
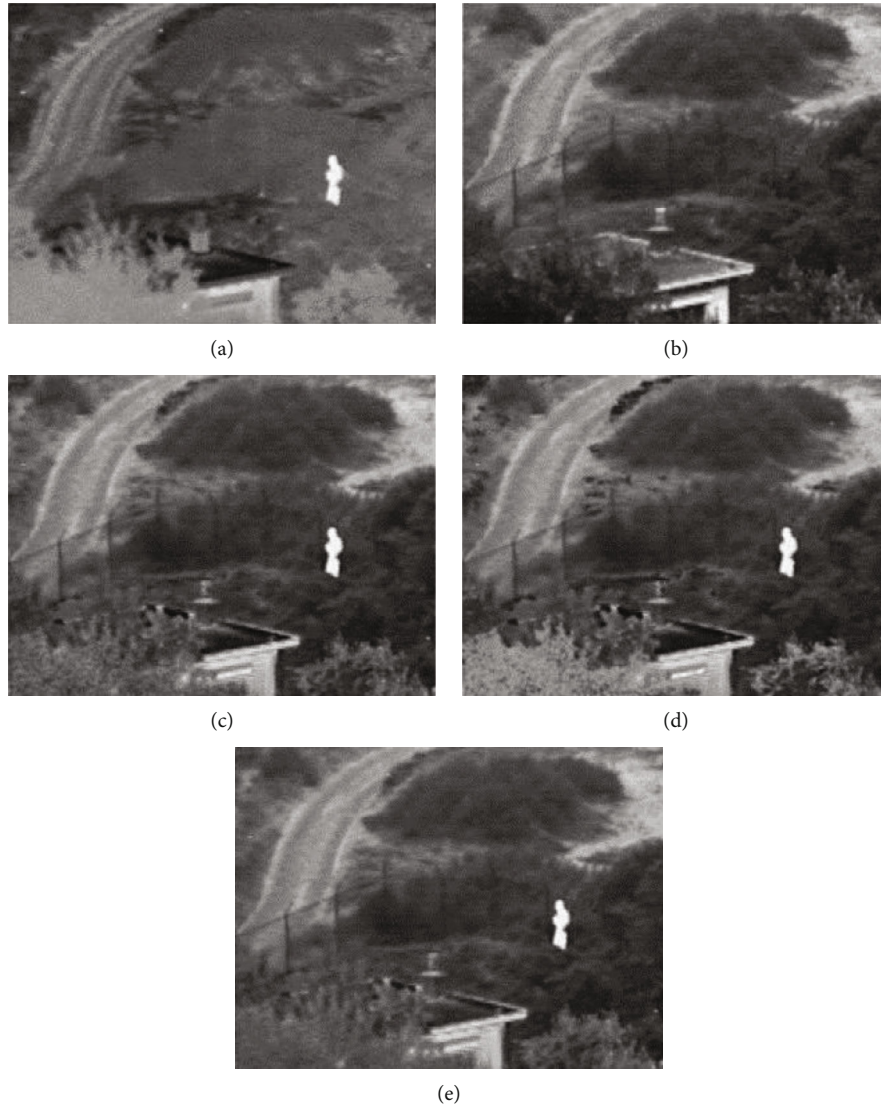
FIGURE 10: Comparison of fusion result based on UN-CAMP images ((a) infrared light sample; (b) visible light sample; (c) feature-level fusion; (d) original TLBO algorithm; (e) improved TLBO algorithm).

in entropy, spatial resolution, and interactive information, and they also have slightly lower standard deviation and mean gradient is slightly lower than the traditional algorithm, but higher than PCA.

Therefore, the fusion result of contourlet decomposition +PCNN transform has a larger amount of information, stronger contrast, and better visual effect. At the same time, it can effectively avoid grayscale distortion and block effect easily caused by the fusion between forest infrared and visible images while avoiding the influence of noises. Therefore, compared with common pixel-level fusion algorithms, this algorithm performs better in improving image information and sharpness.

4.2. Feature-Level Image Fusion Results. Compared with the pixel-level image fusion algorithm, the high-frequency domain fusion algorithm proposed in Figure 7 improves the visual effect of the image. In the forest image, the background area has little influence on the target area, which

reduces the block effect and ringing effect behind. The contour is clearer and more information about the target area is retained. Compared with the pixel pole fusion image, the block effect is significantly reduced. As can be seen from Table 3, pixel-level fusion images have the best data in terms of entropy, mean gradient, standard deviation, and spatial frequency, indicating that pixel-level fusion images are better in terms of information content, contrast, and spatial activity. The spatial resolution and interactive information of feature-level fusion images are optimal, which indicates that the image is better than other fusion images in terms of the degree of association with the source image and noise interference prevention.

As can be seen from the table, the result obtained by the feature-level fusion algorithm has a larger amount of information and better visual effect. It effectively avoids grayscale distortion and block effect easily caused by the fusion of forest infrared and visible images while avoiding the influence of noises. Among different evaluation indexes, pixel-level

(a)

(b)

(c)
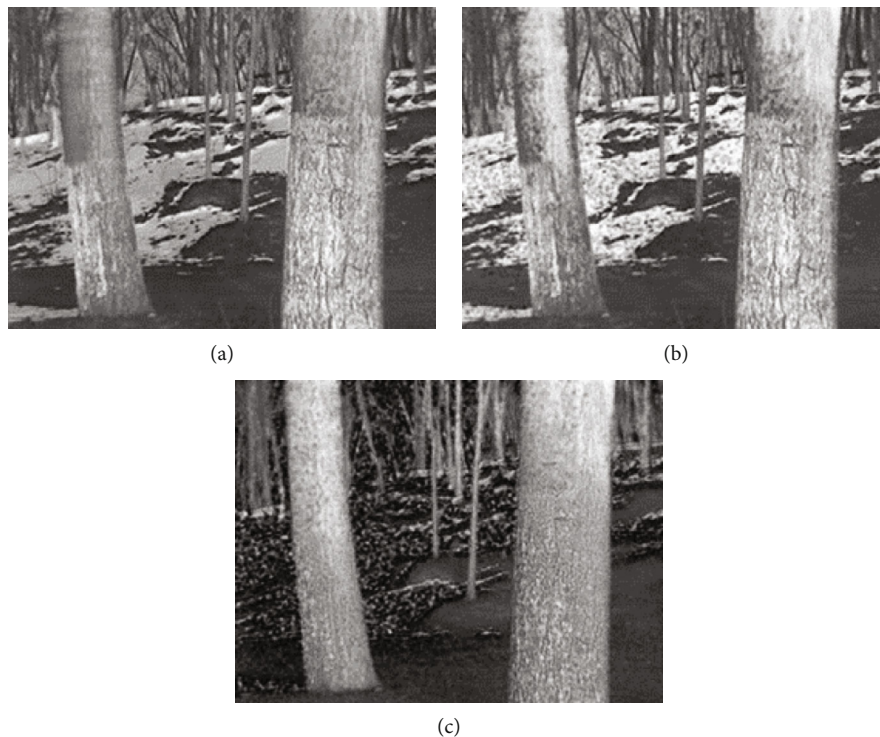
FIGURE 11: Comparison of fusion result based on image samples ((a) feature-level fusion; (b) original TLBO algorithm; (c) improved TLBO algorithm).

TABLE 6: Quality assessment of region-based level and improved TLBO optimization fusion results based on UN-CAMP images.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Region-based level	7.0048	35.2247	4.5529	10.0739	4.8237
TLBO optimization	7.0858	37.6909	5.3584	10.5275	4.8651
Improved TLBO optimization	7.1483	39.9000	5.8673	11.6199	4.9080

TABLE 7: Quality assessment of region-based level and improved TLBO optimization fusion results based on image samples.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Region-based level	7.3981	46.2270	15.4640	29.9097	5.8198
TLBO optimization	7.5121	54.9715	17.4434	33.1374	6.0479
Improved TLBO optimization	7.6921	57.0801	17.6343	29.7725	5.5715

TABLE 8: Quality assessment of region-based level and improved TLBO optimization fusion results based on the other group images.

	Information entropy	Average gradient	Standard deviation	Spatial resolution	Interactive information
Region-based level	7.2378	38.5448	18.2632	31.9562	5.7235
TLBO optimization	7.3910	45.8079	18.8261	32.5719	5.2021
Improved TLBO optimization	7.3979	45.8553	18.6944	32.3447	5.2667

and feature-level fusion results are better, so different fusion methods can be adapted according to different requirements.

4.3. Results of TLBO Algorithm. The feature-level image fusion results of the infrared and visible images mentioned above are shown in Table 4. It can be seen from the table that its entropy value was 7.3981, but was 7.5121 after the optimization of the original TLBO model. Figure 8 shows the comparison between the optimized image and the original feature pole fusion image.

(a)

(b)

(c)
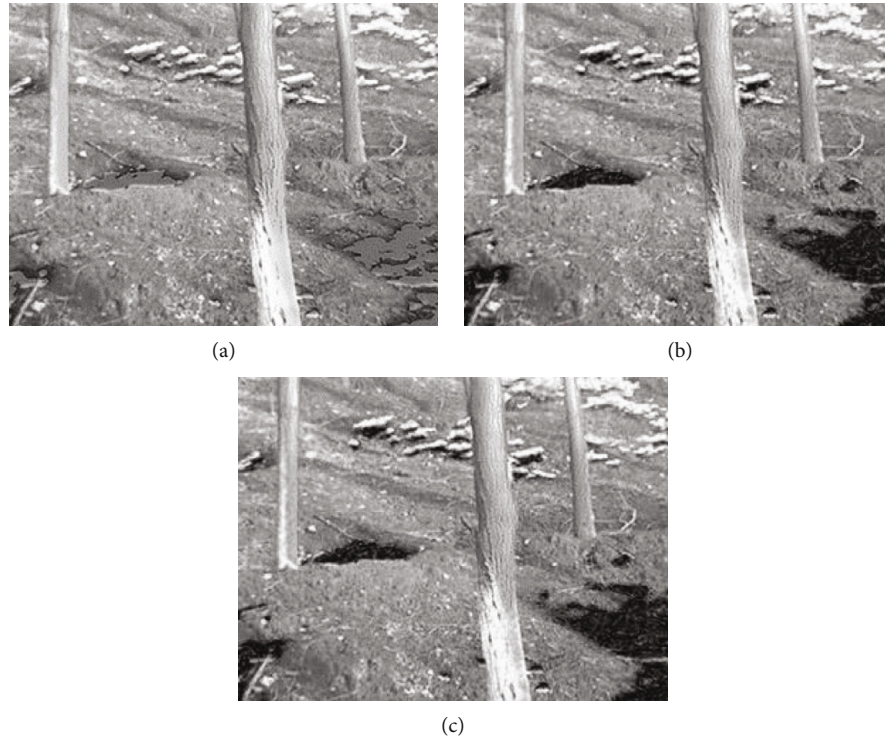
Figure 12: Comparison of fusion result based on the other group images ((a) feature-level fusion; (b) original TLBO algorithm; (c) improved TLBO algorithm).

The entropy value, standard deviation, and mean gradient increased by 1.16%, 7%, and 17.69%, respectively. All the indexes of the optimized image were improved, but not quite significantly.

The feature-level image fusion results of the other group of infrared and visible images are shown in Table 5. It can be seen from the table that its entropy value was 7.2378, but was 7.3910 after the optimization of the original TLBO model. Figure 9 shows the comparison between the optimized image and the fusion image of the original feature pole.

The entropy value, standard deviation, and mean gradient increased by 2.12%, 18.84%, and 3.10%, respectively. Except for interactive information, all the indexes of the optimized image were improved, but not quite significantly. The interaction information represents the relationship between the fusion image and the source image, and the larger the value is, the better the fusion effect is. However, in the fusion image, more infrared image information is needed to obtain a more obvious contour and detailed texture with a better entropy value, so the parameters of interaction information are relatively low.

4.4. Results of Improved TLBO Algorithm. Figures 10(a) and 10(b) are a group of infrared and visible light sample images named UN-CAMP, which have been applied to effect comparison in many domestic and foreign literature on image fusion algorithms. Figure 10(c) is the feature-level fusion image. Figures 10(d) and 10(e) are, respectively, the optimized results of the original TLBO algorithm and the improved TLBO algorithm after the random parameter set-

ting. Table 6 shows the corresponding index evaluation results.

The data in the above table show that all quantitative evaluation indexes of the results after optimization of fusion parameters were improved. Compared with before optimization, the amount of information and spatial activity of images increased by 2.05% and 15.27%, respectively, and the standard deviation and mean gradient of image sharpness and visual effect increased by 13.27% and 28.87%, respectively.

For the infrared and visible image samples selected above, after the same random parameter setting process, the entropy value of the fused image reached the optimal value when the value range of the random parameter R_i and T_f were fixed at [0.4,0.9] and [0.5,1], respectively. Figure 11 shows the feature-level fusion image and the optimized results of the original TLBO algorithm and the improved TLBO algorithm after random parameter setting.

Table 7 shows the evaluation results of corresponding indicators. For this sample, when all entropy values, standard deviation, and mean gradient were improved, the spatial resolution and interactive information data decreased compared with the feature fusion results. Due to complex background information, the improved TLBO algorithm is better in terms of the information content, contrast, and spatial activity of the optimized image. However, the processing results are different from the source image in terms of visual effect due to the excessive influence of background information. In general, the algorithm is quite effective in improving the quality of fusion images and execution efficiency and in

achieving better extraction results of target forest images than other algorithms.

For the infrared and visible images of the other group, after the same random parameter setting, the entropy value of the fused image reached the optimal value when the value range of the random parameter R_i and T_f were fixed at [0.3,0.8] and [0.5,1], respectively. Figure 12 shows the feature-level fusion image, the optimized results of the original TLBO algorithm, and the improved TLBO algorithm after random parameter setting.

Table 8 shows the evaluation results of corresponding indicators. As can be seen from the table, when the entropy value and standard deviation indicators were improved, the spatial resolution and mean gradient data became lower than the original optimization results, but still higher than the feature-level fusion image. In the algorithm, the amount of information and contrast of the optimized image are better. By comparing the results obtained from multiple sets of data, it can be seen that for different image samples, the algorithm has relatively optimized effects in improving the quality of fusion images and execution efficiency and could achieve better extraction results of target forest images than other algorithms.

5. Discussion and Conclusions

In the pixel-level image fusion algorithm research, the pulse coupled neural network model relying on contourlet transform is applied to avoid block effect and grayscale distortion caused by the fusion of infrared and visible images. Given the significant difference in gray level between infrared and visible forest images, a reasonable threshold value is selected in the low-frequency domain fusion processing. The points with different output pulse signals are treated differently, and the fusion rules are explicitly formulated. Thus, grayscale distortion and block effect are avoided, but the quality of the fusion image can be effectively improved, and all evaluation indexes of the image can be improved to some extent. In the research of feature-level image fusion algorithms, the PCNN model was used to eliminate the influence of noise in path optimization. The segmentation consequences are sufficient to meet the needs of feature-level fusion research even though they are disturbed by confusable issues. Based on the fuzzy logic rules, the fusion rules in the low-frequency domain are formulated by calculating the degree of dissimilarity between corresponding points of source images. The fusion rules in the high-frequency domain are determined by combining the image segmentation results. While ensuring the visual effect of the fused image, the detailed characteristic information of the target region in the image was displayed, making the research on image fusion a more targeted and purposeful algorithm into the algorithm to improve further the local search ability of the algorithm [24, 25]. The experimental results show that the feature-level image fusion algorithm ensures image quality, achieves the detailed display of the tree target area in the image, and improves quality evaluation indexes. Compared with the pixel-level fusion results, the tree texture obtained by this method is more evident, with more apparent edges.

In the research of feature-level image optimization, the teaching learning-based optimization (TLBO) parameter optimization algorithm is introduced to optimize the fusion coefficient in the fusion process to improve the fusion image's index data. In order to obtain better image results, the optimal parameter combination for different image groups to achieve the optimal effect by setting the value range of random parameters in the TLBO model and various quantitative evaluation indexes of fused images was improved. Pixel-level and feature-level fusion algorithms have appropriate advantages for different occasions. Pixel-level fusion has advantages in improving image information and sharpness, but it takes twice as long to process information as feature-level fusion. Feature-level fusion has a broader application space in forestry intelligent information detection as it can highlight the target area and reduce computation. The setting of algorithm parameters has an important influence on its optimization ability. The teaching factor of the basic TLBO algorithm varies only, which affects the optimization performance of the algorithm. Therefore, an improved TLBO optimization algorithm is proposed to design the teaching factor by segmenting strategy to process the image. Experimental results verify the algorithm's effectiveness, which has good searching ability and fusion image quality. In the future, this proposed method will be extended to theoretical research and practical applications including time-serial prediction and pattern recognition [16, 26–28].

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare no conflict of interest.

Authors' Contributions

Conceptualization was done by Jinghua Wang and Lei Yan; methodology was done by Jinghua Wang; software was done by Jinghua Wang; validation was done by Jinghua Wang; formal analysis was done by Lei Yan and Fan Wang; investigation was done by Lei Yan and Fan Wang; resources was done by Lei Yan; data curation was done by Jinghua Wang and Lei Yan; writing—original draft preparation was done by Jinghua Wang; writing—review and editing was done by Jinghua Wang and Shulin Li; visualization was done by Jinghua Wang and Shulin Li; supervision was done by Lei Yan; project administration was done by Lei Yan; funding acquisition was done by Lei Yan. All authors have read and agreed to the published version of the manuscript.

Acknowledgments

References

[1] J. Kong, H. Wang, X. Wang, X. Jin, X. Fang, and S. Lin, "Multi-stream hybrid architecture based on cross-level fusion strategy for fine-grained crop species recognition in precision agriculture," *Computers and Electronics in Agriculture*, vol. 185, article 106134, 2021.

[2] J. Kong, C. Yang, J. Wang et al., "Deep-stacking network approach by multisource data mining for hazardous risk identification in IoT-based intelligent food management systems," *Computational Intelligence and Neuroscience*, vol. 2021, Article ID 1194565, 16 pages, 2021.

[3] M. I. Daily, T. Farr, and C. Elachi, "Geologic interpretation from composited radar and Lansat imagery," *Photogrammetric Engineering and Remote Sensing*, vol. 45, no. 8, pp. 1109–1116, 1979.

[4] A. Toet, "Image fusion by a ratio of low-pass pyramid," *Patten Recognition Letters*, vol. 9, no. 4, pp. 245–253, 1989.

[5] R. A. Eggleston and C. A. Kohl, "Symbolic fusion of MMW and IR imagery," *Proceedings of SPIE*, vol. 1003, pp. 20–27, 1988.

[6] A. Toet, J. K. Ijspeert, A. M. Waxman, and M. Aguilar, "Fusion of visible and thermal imagery improves situational awareness," *Proceedings of SPIE on Enhanced and Synthetic Vision*, vol. 3088, pp. 177–188, 1997.

[7] L. P. Pappas, P. Malik, and M. Styner, "New method to assess the registration of CT·MR images of the head," *Medical Imaging 2004: Image Processing*, vol. 5370, pp. 129–136, 2004.

[8] X. B. Jin, W. Z. Zheng, J. L. Kong et al., "Deep-learning forecasting method for electric power load via attention-based encoder-decoder with Bayesian optimization," *Energies*, vol. 14, no. 6, p. 1596, 2021.

[9] Y. Liu, S. Liu, and Z. Wang, "Multi-focus image fusion with dense SIFT," *Information Fusion*, vol. 23, pp. 139–155, 2015.

[10] D. M. Bulanona, T. F. Burksa, and V. Alchanatisb, "Image fusion of visible and thermal images for fruit detection," *Biosystems Engineering*, vol. 103, no. 1, pp. 12–22, 2009.

[11] Y. Lei, D. Xiaokang, K. Jianlei, Y. Zheng, and L. Jinhao, "Parameters optimization algorithms for improving the performance of obstacles identification in forest area," *INMA-TEH-Agricultural Engineering*, vol. 40, no. 2, pp. 43–52, 2013.

[12] Y. Lei, D. Xiaokang, Y. Zheng, K. Jianlei, and L. Jinhao, "A novel identification method of obstacles based on multisensor data fusion in forest," *Sensors & Transducers*, vol. 155, no. 8, p. 155, 2013.

[13] C. Pohl and J. L. Van Genderen, "Review article multisensor image fusion in remote sensing: concepts, methods and applications," *International Journal of Remote Sensing*, vol. 19, no. 5, pp. 823–854, 1998.

[14] G. Stephane and A. Matllat, "A theory for multiresolution signal decomposition: the wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, 1989.

[15] M. Do and M. Vetterli, "Contourlets: a directional multiresolution image representation," *International Conference of Image Processing Proceedings*, vol. 1, pp. 357–369, 2002.

[16] X. B. Jin, W. Z. Zheng, J. L. Kong et al., "Deep-learning temporal predictor via bi-directional self-attentive encoder-decoder framework for IOT-based environmental sensing in intelligent greenhouse," *Agriculture*, vol. 11, no. 8, p. 802, 2021.

[17] Y. Y. Zheng, J. L. Kong, X. B. Jin, X. Y. Wang, T. L. Su, and M. Zuo, "CropDeep: the crop vision dataset for deep-learning-based classification and detection in precision agriculture," *Sensors*, vol. 19, no. 5, p. 1058, 2019.

[18] Y. Y. Zheng, J. L. Kong, X. B. Jin, X. Y. Wang, T. L. Su, and M. Zuo, "Probability fusion decision framework of multiple deep neural networks for fine-grained visual classification," *IEEE Access*, vol. 7, pp. 122740–122757, 2019.

[19] J. Teng, S. Wang, J. Zhang, and W. Xue, "Fusion algorithm of medical images based on fuzzy logic," in *2010 Seventh International Conference on Fuzzy Systems and Knowledge Discovery*, pp. 546–559, Yantai, China, 2010.

[20] W. Cai and Z. Wei, "Pii GAN: generative adversarial networks for pluralistic image inpainting," *IEEE Access*, vol. 8, pp. 48451–48463, 2019.

[21] R. V. Rao, V. J. Savsani, and D. P. Vakharia, "Teaching-learning-based optimization: a novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, 2011.

[22] H. Jin and Y. Wang, "A fusion method for visible and infrared images based on contrast pyramid with teaching learning based optimization," *Infrared Physics & Technology*, vol. 64, pp. 134–142, 2014.

[23] Y. Han, Y. Cai, Y. Cao, and X. Xu, "A new image fusion performance metric based on visual information fidelity," *Information Fusion*, vol. 14, no. 2, pp. 127–135, 2013.

[24] B. Mandal and P. K. Roy, "Optimal reactive power dispatch using quasi-oppositional teaching learning based optimization," *International Journal of Electrical Power & Energy Systems*, vol. 53, pp. 123–134, 2013.

[25] R. V. Rao and V. Patel, "Comparative performance of an elitist teaching-learning-based optimization algorithm for solving unconstrained optimization problems," *International Journal of Industrial Engineering Computations*, vol. 4, no. 1, pp. 29–50, 2013.

[26] X.-B. Jin, W.-T. Gong, J.-L. Kong, Y.-T. Bai, and T.-L. Su, "PFVAE: a planar flow-based variational auto-encoder prediction model for time series data," *Mathematics*, vol. 10, no. 4, p. 610, 2022.

[27] X.-B. Jin, J.-S. Zhang, J.-L. Kong, Y.-T. Bai, and T.-L. Su, "A reversible automatic selection normalization (RASN) deep network for predicting in the smart agriculture system," *Agronomy*, vol. 12, p. 591, 2022.

[28] J.-L. Kong, H.-X. Wang, C.-C. Yang, X.-B. Jin, M. Zuo, and X. Zhang, "Fine-grained pests and diseases recognition via spatial feature-enhanced attention architecture with high-order pooling representation for precision agriculture practice," *Agriculture*, vol. 2022, article 1592804, 2022.

Research Article

Intelligent Sensing and Computing in Wireless Sensor Networks for Multiple Target Tracking

Xinyu Zou [ID],[1] **Linling Li,**[2] **Hao Du,**[2] **and Longyu Zhou** [ID][2]

[1]*IT Center, West China Hospital of Sichuan University, Chengdu 610041, China*
[2]*School of Information & Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China*

Correspondence should be addressed to Longyu Zhou; zhoulyfuture@outlook.com

With sixth generation (6G) communication technologies, target sensing can be finished in milliseconds. The mobile tracking-oriented Internet of Things (MTT-IoT) as a kind of emerging application network can detect sensor nodes and track targets within their sensing ranges cooperatively. Nevertheless, huge data processing and low latency demands put tremendous pressure on the conventional architecture where sensing data is executed in the remote cloud and the short transmission distance of 6G channels presents new challenges into the design of network topology. To cope with the above difficulties, this paper proposes a new resource allocation scheme to perform delicate node scheduling and accurate tracking in multitarget tracking mobile networks. The dynamic tracking problem is formulated as an infinite horizon Markov Decision Process (MDP), where the state space that considers energy consumption, system responding delay, and target important degree is extended. A model-free reinforcement learning is applied to obtain satisfied tracking actions by frequent iterations, in which smart agents interact with the complicated environment directly. The performance of each episode is evaluated by the action-value function in search of the optimal reward. Simulation results demonstrate that the proposed scheme shows excellent tracking performance in terms of energy cost and tracking delay.

1. Introduction

With the new technologies in 6G, ultralow latency for huge data transfer dramatically enlarges the target sensing application field. Mobile target sensing in Wireless Sensor Networks (WSNs), which is dedicated to monitoring the supervisory field in the military and civilian fields for security, is a kind of special Internet of Things (IoT) application [1, 2]. In the military field, this application can be applied in countering terrorism, unmanned aerial vehicle (UAV) navigation, and space exploration, etc. Relatively speaking, no-flying zone monitoring of airports and information collecting also used this application for security and data acquirement [3]. Specially, the sensor nodes detect whether the suspicious target intrudes into the ground for target sensing and establish a communication connection with the sink node. Controlling data is issued for scheduling potential sensors to execute sensing task on the sink node. Consequently,

the sink node is regarded as a controller with sufficient computing capacity for the sensor nodes.

However, optimal sensing time delinquency becomes common due to high latency response in the traditional centralized architecture. Specifically, computing time is obviously increased when the massive data is required from sensor nodes in WSNs. New technologies of 6G, such as terahertz channels, may provide enough bandwidth for the huge data transmission [4], but the unstable and short transmission link put new constraints on the topology control of the sensor network [5]. Furthermore, redundant data has also emerged because more than one sensor node can detect the same target with target nodes moving and the coverage areas of sensor nodes could have overlapping in the practical scenario. Besides, transmission congestion and data queue still exist in the MTT-WSNs. As a consequence, sensing time moves backwards in turn, missing the best sensing opportunity.

In addition, the low energy consumption of WSNs has always been the focus of attention, with the characteristics of battery power. Generally speaking, energy cost is mainly divided into transmission cost and sensing cost, in which sensing cost is more than transmission cost when the detected target nodes break away from control. How to schedule the proper sensor nodes for saving energy cost is the other exposed challenge. In order to improve the target sensing performance of WSNs, many scholars and engineers have done a lot of researches.

In recent years, satisfactory results have been obtained by vast studies on disparate sensing performances. In terms of a compromise solution between sensing quality and energy efficiency, the authors of [6] design a sensing scheme named "t-tracking," which enhances quality of sensing by increasing network throughput. Sensing performance is optimized by Luo et al. [7] in improvement of localization by a cooperative localization scheme. Tracking accuracy is promoted by the combination of the off-line phase and the on-line phase, which largely offsets the slower response time. Edge computing provides a new paradigm for good tracking performance. The authors of [8, 9] introduced a similar cooperative tracking algorithm based on edge computing technology with minimum energy consumption, in which the computing device is located near the sensor node. Wan et al. [10] propose a joint Doppler-angle estimation solution for improving multitarget tracking efficiency.

In general, the multitarget tracking of WSNs faces main challenges as follow:

(i) When targets with different important degrees invade the monitoring area, it is difficult to accurately identify the targets' threat level. In some important monitoring areas, the highest important degree targets may be missed

(ii) Task-oriented WSNs have the problem of unreasonable bandwidth utilization. Low important degree intrusion targets may consume limited bandwidth, so communication for monitoring high important degree targets cannot be guaranteed

(iii) The limited mobility and coverage ability of sensor nodes may lead to the targets with high important degree be missed or misjudged. As a result, the tracking accuracy may be not satisfactory

(iv) The high-intensity computing consumption of the central server leads to high tracing delays and cannot meet the requirements of real-time control. Effective feedback cannot be guaranteed

Motivated by the observations, we propose a distributed multitarget tracking scheme with intelligent edge device in this paper, which is different from those of target tracking system with edge computing or localization algorithms [11]–[12]. On the one hand, compared with the edge computing system, the offloading strategy needs a joint decision, including whether to schedule, which sensor to schedule, and which target to track, which makes the offloading strat-

egy more challenging. On the other hand, compared with the traditional location algorithm, the response time can be reduced rapidly by locating the distributed edge devices. The optimal target is transferred from minimum energy cost or tracking delay to joint consideration of energy cost and tracking performance. The main contributions are summarized as follows:

(i) We propose a new distributed multitarget tracking architecture, where real-time scheduling is enhanced by requiring computing results from resource caches. Besides, we improve the tracking efficiency by introducing mobile cloud server providing resources for sink nodes. Missing probability is reduced with the assistance of collaborative perception among sensor nodes

(ii) Based on our proposed architecture, the performance multitarget tracking is improved with joint consideration of system cost, response time, and important degree of targets. Minimum energy cost with low tracking delay is transferred as MDP with free model representation which depends on the current system state

(iii) Self-adaption is proposed by an elastic mechanism, which optimizes energy consumption and response time. Based on the MDP model, we establish the action-value function to evaluate the scheduling strategies with the energy cost and tracking delay of sensor nods. A deep Q-learning Network (DQN) algorithm is utilized for optimizing real-time scheduling strategies to adapt to different scenarios for different purposes, such as targets with key tasks or high mobility. Simulation results demonstrate that our intelligent cooperative scheme shows good tracking performance according to the self-adaption scheme

The rest of this paper is organized as follows. System model is given by Section II. The scheduling strategy is analysed by Section III. Section IV presents the simulation evaluation. Finally, Section V concludes this paper.

2. Related Work

Energy consumption, delay, coverage, sensor deployment, and accuracy are crucial factors in sensor networks about target tracking [13]. In recent years, many researchers have done a lot of work to address these challenging issues [14]–[15].

To begin with, some researches have studied the problem of tracking coverage. The purpose of [16] is to provide an effective coverage method for applications such as target location, environmental monitoring, and vehicle target tracking. Some intelligent algorithms, such as neural networks, adaptive distributed filtering, and fuzzy framework,

are used to obtain the optimal filter. A new adaptive Kalman estimation method based on distribution consistency is proposed [17]. The average inconsistency of the optimal filter gain and the estimated value are considered in the filter design. Optimal Kalman gain is obtained by minimizing mean square estimation error to estimate the states of the target more accurately. In order to get better filter performance, an adaptive consistency factor is used to adjust the optimal gain. In the information exchange of filters, dynamic clustering selection and two-level hierarchical fusion structure are used to obtain more precise estimation [18].

Meanwhile, the architecture of WSNs is extensively studied. One of the most popular applications in WSNs is vehicle tracking. Generally speaking, vehicle tracking involves edge computing, edge intelligence, etc [19]. Many researches focus on improving the tracking accuracy and energy saving of target vehicles on WSNs. A decentralized vehicle tracking strategy is proposed to dynamically adjust the active area, improve the energy-saving effect, and tracking accuracy. [20] introduces novel mobile sensor networking architecture for a swarm of microunmanned vehicles (MAVs) using software defined network (SDN) technology. The proposed architecture was aimed at enhancing the performance of user/control plane data transmission between MAVs. In [21], the authors propose a novel system architecture and MAC protocol, which includes how to select the target cooperative sensor node and which sensing data should be retransmitted by using the decided cooperative sensor node.

Moreover, due to the extensive research on machine learning, lots of researches have upgraded and extended the algorithm, which can be widely used in WSNs [22]. In [23], they proposed a double time scale Q-learning algorithm with function approximation to alleviate the curse of dimension problems. Although all of the above algorithms alleviate the state explosion problem, it is necessary to solve the action explosion problem to obtain the scalable solution. The edge intelligent computing of WSNs has also been widely studied, such as the basic wireless cache networks, in which the source server is connected to the base station (BS) supporting cache, which serves multiple requesting users [24]. In order to solve the problem of how to improve the cache hit rate under the condition of dynamic content popularization, a dynamic content update strategy based on deep reinforcement learning is proposed. In [25], the authors considered a WSN with a great number of sensors and multiple clusters. Each cluster has a cluster head and several cluster members, and the sensors are distributed randomly. Cluster members only measure environmental parameters like humidity, atmospheric pressure, and temperature. The cluster head is responsible for fusing the measurement data from the cluster members and forwarding the data to the base station through single hop or multihops.

Furthermore, with the development of WSNs and UAV technologies, which is popularly used in target tracking, the enhancement of coverage, the comprehensive utilization of data resources, and the stability of network systems, especially the cooperative network composed of WSNs and UAV, are expected to provide immediate long-term benefits in military and civil fields. Previous work mainly focused on how to use UAV assisted WSNs for sensor and data collection, or use a single data source for target location in the monitoring system, but the potential of multiple UAV sensor networks has not been fully explored. In [26], the problem of target tracking for a class of wireless positioning system with degradation measurement and quantization effect is investigated. In order to track the state of the object as accurately as possible, a recursive filtering algorithm is proposed. First, an upper bound of filter error covariance is derived, and then, the upper bound is minimized by properly designing the filter gain at each sampling time. [27] proposed a new network platform and a system structure for multiple UAVs' cooperative monitoring, including designing idea of cooperative resource scheduling and tasking allocation scheme for multiple UAVs. Due to the complexity of the monitoring area, [28] discussed the establishment of a suitable algorithm based on machine learning.

3. System Model

In this section, a state-action model is established for the distributed multitarget tracking system. State space is added which incorporates requiring latency, queue length, and different important degree of targets. Each indicator is specifically designed in the following communication model and request model.

As shown in Figure 1, the efficient computing resources enabled MTT network is made up of one UAV-assisted mobile station, N target nodes, M sensor nodes, and K sink nodes. Sink nodes are selected from the set M, i.e., $K \subset M$. Each sensor node is an energy-limited device with powered battery. Let $\mathcal{N} = \{1, 2, \cdots N\}$ denotes the set of target nodes and $\mathcal{M} = \{1, 2, \cdots M\}$ denotes the set of sensor nodes, respectively. Let $t \in \{1, 2, \cdots, T\}$ denotes the discrete time for tracking. Sensor nodes perceive targets each time and report to sink nodes which are equipped with a certain amount of computing resources. The computing recourses are set as $C \subset N$. Sink nodes schedule proper sensor nodes to execute tracking tasks under the assistance of UAV. We assume that each sink node owns corresponding network information, in which UAV obtains the information via each fixed period by information sharing.

3.1. Analysis of Communication Model and Action. In this subsection, the communication model and action space are analysed. We assume that arrival of target nodes is on the basis of an independently identically distribution (IID). Once the probability happens, energy consumption is produced at time t. Hence, we depict the energy consumption as $e_{r,t}$. Requiring to be scheduled is real time for sensor node m, and the overhead of fetching transmission should be considered as follows:

$$e_{r,t} = \lambda_{elec} \times 2 + \lambda_{amp} \times d^2, \tag{1}$$

where λ_{elec} is the circuit consumption and λ_{amp} is the gain consumption of amplifier consumption of one bit. Downlink of 1 bit can realize scheduling strategy, i.e., 0 is prohibition
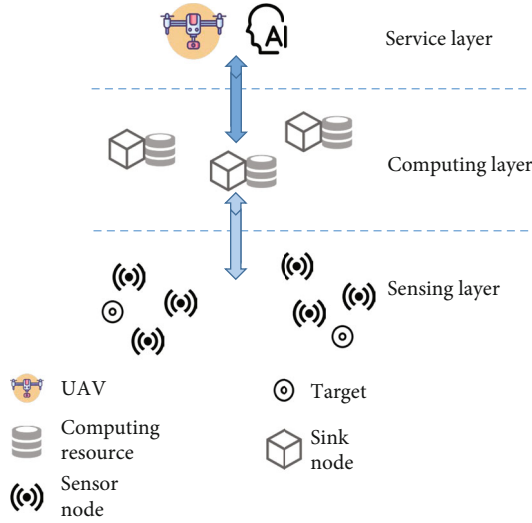
FIGURE 1: System components and transmitting modes based collaboration and scheduling solution in MTT-WSNs.

otherwise is permission. d is the distance between the sensor and the sink node.

Sink node provides a proper scheduling scheme to track wisely, and the scheduled probability of tracking target n at time t is denoted as $p_{m,n,t}$. The energy consumption of the sensor node m as a candidate at time t is denoted as $e_{s,t}$. In the dynamic self-recognized network, it is noteworthy that the information of the sensor node m is absent for the region r when the sensor node moves from the region h to the region r. However, the corresponding data can be obtained from UAV and the fetching cost is associated. Let b_m denote the cost between UAV and the sink node, the fetching cost is written as follows:

$$F_m = 1(m \notin C^m)b_m, C^m \subset \mathscr{C}, \tag{2}$$

where $1(*)$ is the indicator function. If $*$ is generated, cost will be incurred, otherwise, cost is zero.

In the multitarget tracking network, the energy of the sensor nodes and the relative distance between each target node are considered for evaluating the potential tracking capacity. Let $a_t \in \mathscr{A} \triangleq \{M \cup 0\}$ denotes the decision-making variable for real time tracking at time t, where in $a_t = 0$ denotes no decision for sink node, $a_t = m$ denotes sensor node m and is scheduled at time t, and $\mathscr{A}$ is the action space.

3.2. Analysis of Request and Delay Model.
In this subsection, the request model and delay model are analysed, incorporating the design of the request queue and delay. At each time t, a sensor node can only receive information of one target node. Each sensor node reports to the corresponding sink node. When tracking target n, other sensor nodes may receive broadcasting data of target n. Starting at time t, sink nodes requested by the set of sensor nodes is set as $\mathscr{M}^b_{n,t}$, and $\mathscr{M}^b_{n,t} \subset \{\mathscr{M} \cup 0\}$. When the number of targets invading the monitoring area is none, $\mathscr{M}^b_{n,t} = \varnothing$. $\mathscr{M}^d_{n,t}$ is the set of sensor nodes when requesting sink nodes to track target n at time t,

and the dynamic renewal process is described as follows:

$$\mathscr{M}^b_{n,t+1} = \mathscr{M}^b_{n,t} \cup \mathscr{M}^d_{n,t}, \tag{3}$$

$$\left| \mathscr{M}^b_{n,t+1} \right| = \begin{cases} \left| \mathscr{M} \setminus \mathscr{M}^d_{n,t} \right|, & if \, r_t = n \\ 0, & otherwise \end{cases}, \tag{4}$$

where $r_t = n$ indicates the sink nodes receive request from sensor nodes to track target node n. Otherwise, there is no request.

At each time t, the queue for requesting sink nodes at the beginning of the next time $t + 1$ is based on the last time. If sensor nodes have no idea for target node m, the number of requests should be computed with a view to number of requests at time t during the next time $t + 1$. The size of request unit length is $\mu_{m,n,t}$, and the request queue length exception $R_{n,t+1}$ at time $t + 1$ is represented as follows:

$$R_{n,t+1} = \sum_{m=1}^{M} \mu_{m,n,t}. \tag{5}$$

The scheduling set is explored as a subset of the request queue. In case of $R_{n,m,t} = 0$, the scheduling set is empty. Otherwise, the scheduling set belongs to the request queue. The $N \times M$ state matrix $\mathscr{R}_t$ is constructed, and $\mathscr{R}_t \in \mathbf{R}$, where $\mathbf{R}$ is the state space of the request queue. The independence between sensor nodes leads the scheduling policy not to be related to the request probability but only related to the nonempty request queue.

At the beginning of time t, sensor node m transmits its own information to the sink node and is traversed according to the corresponding node ID which is stored in the computing resources. After that, the policy notifies the sink node to make processing. The fetching transmitting delay is analysed and requesting nodes with probability $p_{n,m,t}$ is scheduled. The penalty of delayed sensor nodes is generated in a subset of request nodes. Heterogeneous target nodes with different delay constraints are considered to be associated with different delay penalties. At time t, the delay constraint range for target node n is written as $I_{n,t} \in I_n \in [0, \bar{I}_n]$, where the maximum delay is $\bar{I}_n$. $\mathbf{I}$ denotes the delay state space and $\bar{I}_n \in \mathbf{I}$. Therefore, the dynamic delay is described as

$$I_{n,t+1} = \begin{cases} 0, & if \, \mu_{n,t} = 0 \\ \min \left[I(p_{n,t}), \bar{I}_{n,t+1} \right], & otherwise \end{cases}, \tag{6}$$

where $I(*)$ is a utility function of the delay, which mainly includes transmitting delay, and fetching delay between the sink nodes and the UAV.

In the MTT-WSNs, penalties will increase by approaching the upper bound of delay constraint. In order to avoid high transmission delays and interminable request queues, let $\sigma_{i,t}, \forall i \in N$ denotes the important degree of target node m which also expresses the degree of not missing target n. Important degrees for target node are denoted as $\sigma_t = [$
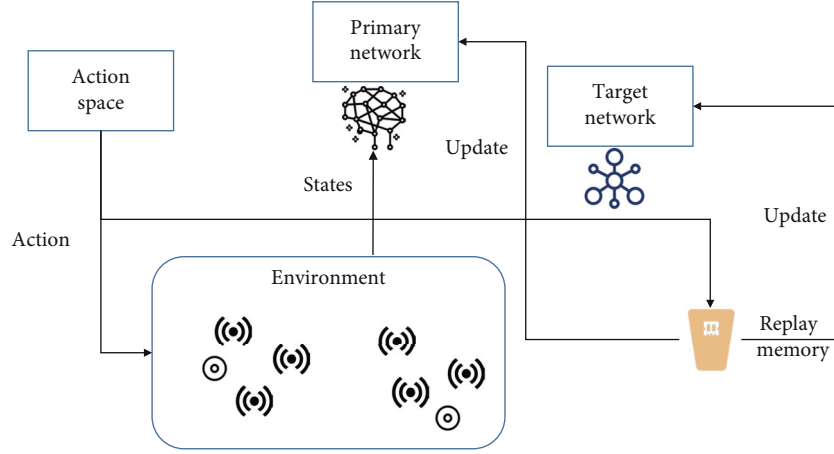
FIGURE 2: DQN-based scheduling strategy and interacting process between the agent and MTT-WSNs.

Initialize the replay memory D to capacity N and action- value Q with random weights
Input: set vector s_t, step number N, discount factor γ, learning rate α, the number of step M
1: for each interaction with environment episode =1, M
   do
2:   for t =1, T do
3:      With probability $?$ select a random action α_t
4:      The corresponding penalty D is computed and stored in memory;
5:      Get a batch of samples from replay memory, and compute difference function based on (17);
6:      Calculate the gradient of weight α;
7:      Updating: according to (18);
8:      if step = N then
9:      $Q \leftarrow Q'$
10:    end if
11:   end for
12: end for
Output: Q

ALGORITHM 1: DQN-based scheduling strategy.

TABLE 1: Simulation parameters of MTT-WSNs' environment.

Parameter description	Value
Area of each region	100 m × 100 m
Number of target nodes	2
Velocity of target nodes	1 m/s
Number of sensor nodes	40
Velocity of sensor nodes	2 m/s
Maximum tolerate delay of target nodes	5 s, 6 s
Primary energy of each sensor nodes	40 J
Primary coordinate of target nodes	(0 m, 30 m), (0 m, 60 m)
Energy consumption for static nodes	0.2 J/unit time
Energy consumption for moving nodes	0.8 J/unit time

$\sigma_{1,t}, \sigma_{2,t}, \sigma_{n,t}]$, and $\sigma_t \in \boldsymbol{\sigma}$, where $\boldsymbol{\sigma}$ belongs to the following state space.

As consequence, we describe the system state as $\mathbf{S} = \mathbf{R} \times \mathbf{I} \times \boldsymbol{\sigma}$ by integrating the state space of delay, request queue, and important degree. Meanwhile, we take a single self-organized network in consideration to simplify the complexity for the reason that the same policy can be realized in different self-organized regions. The state space S is reduced to $\mathbf{S}_1 = \mathbf{R}_1 \times \mathbf{I}_1 \times \boldsymbol{\sigma}_1$. At time t, the state space is given by

$$\mathbf{S}_1 = \left\{ s_t = \mathbf{R}_{1,t} \times \mathbf{I}_{1,t} \boldsymbol{\sigma}_{1,t} \big| p\left(I_{1,m,t} \times \sigma_{1,n,t} \big| R_{1,m,t} \right) \right\}, \quad (7)$$

where delay constraint only appeared when request queue is nonempty.

4. Scheduling Strategy for MTT-WSNs

The exploring strategy is analysed in this section by means of self-adaptation deep learning, in which the interacting learning is designed between a smart agent and a complex monitoring environment.

4.1. Problem Formulation. Given a state space $s \in \mathbf{S}_1$, the action, which is evaluated with a corresponding penalty or reward, resulted from the state space. Policy π is a function
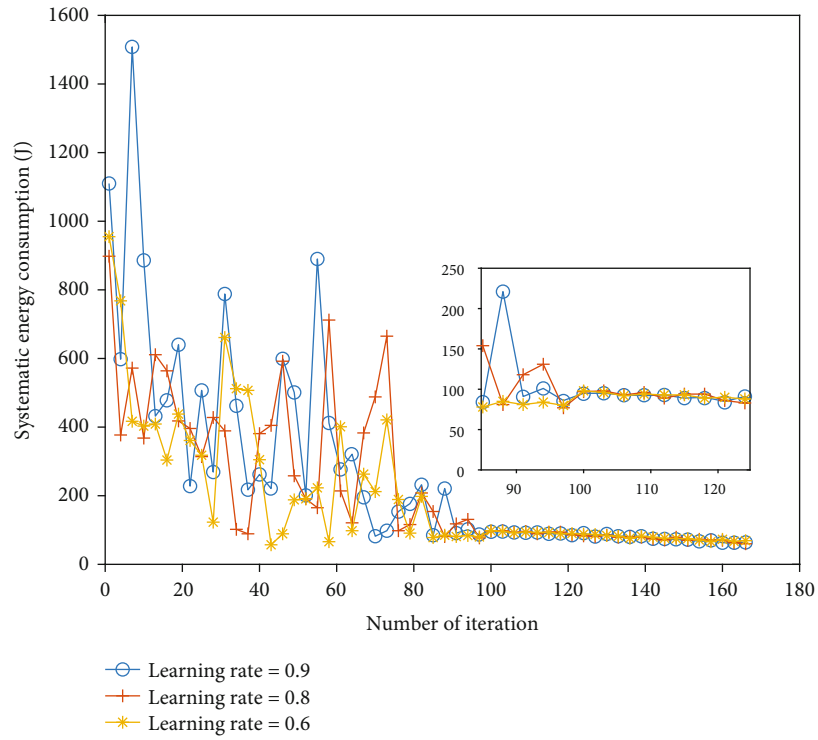
FIGURE 3: The systematic energy consumption under the condition of $\xi_1 = \xi_2 = 0.5$.
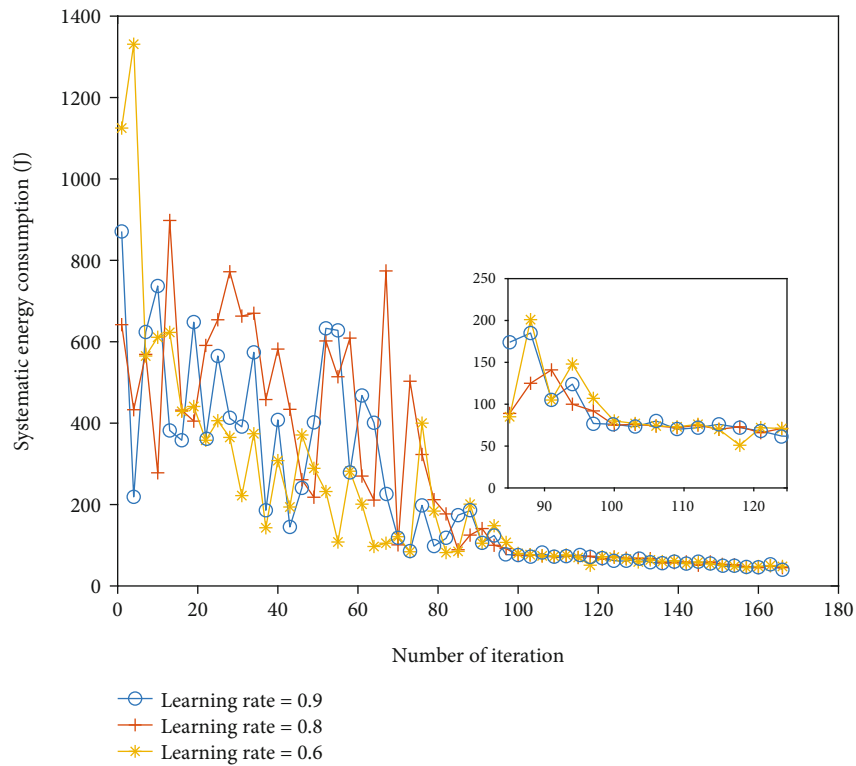
FIGURE 4: The systematic energy consumption under the condition of $\xi_1 = 0.9, \xi_2 = 0.1$.
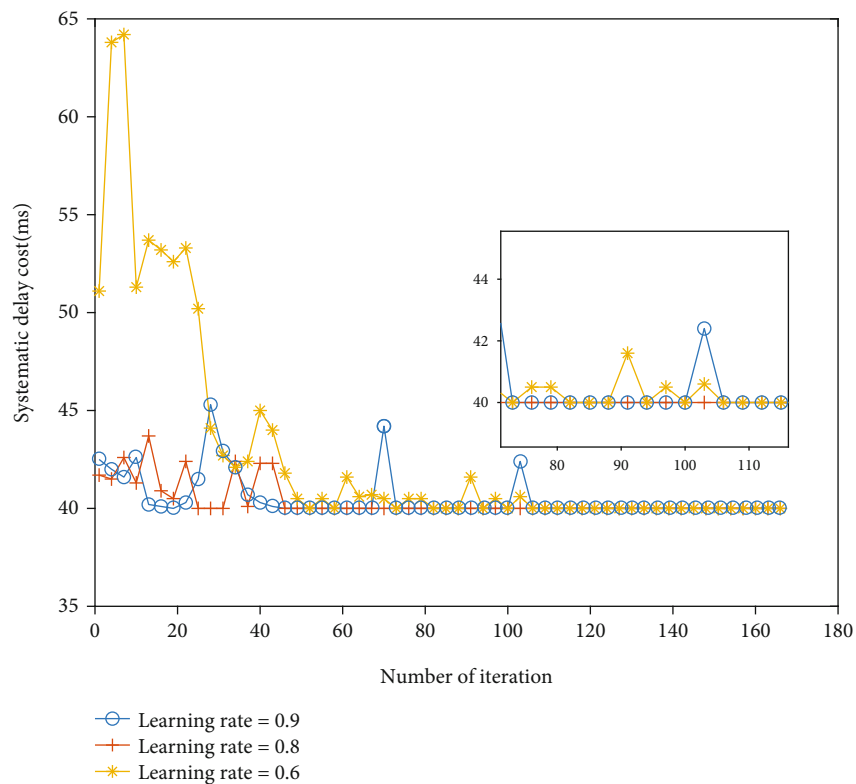
FIGURE 5: The systematic delay cost under the condition of $\xi_1 = 0.9, \xi_2 = 0.1$.
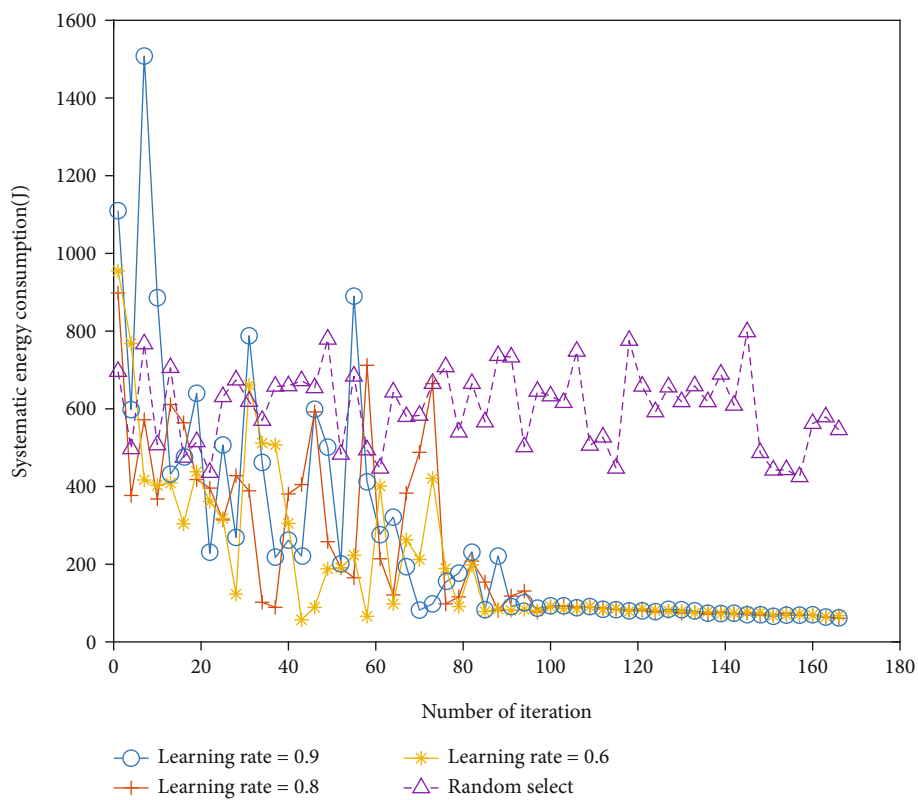
FIGURE 6: The comparison with RSN under the condition of $\xi_1 = \xi_2 = 0.5$.

mapping from state space to action space, which is given by

$$s \xrightarrow{\pi} \mathcal{A}. \tag{8}$$

In the MTT-WSNs, the optimal solution is unavailable by using existing deep learning algorithms such as deep Q-learning for the state space $\mathbf{S}_1$ with multiple dimensions. Thus, the problem is solved by training in the neural network instead of finding in the Q-table. The solution can cope with the curse of dimension because of multiple tensors of the neural network. Specifically, multidimensional state space as input is learned to obtain a policy through deep neural network with weighted values. Action is obtained by the branch of policy and is executed in a dynamic and complex environment, which generates a new state and the corresponding reward or penalty. If this evaluation is a positive reward, a new state value is set as the next input. Otherwise, this action will be punished. Corresponding reward or penalty is responsible for adjusting weight parameters for the deep neural network [23].

In a practical mobile target tracking system, energy consumption can be represented for sensor node m at time t by the (state and action) pair. For a given target node n, the systematic energy consumption is written as

$$E(\pi) = \sum_{t=1}^{T} E_t(s_t, a_t), \tag{9}$$

$$E_t(s_t, a_t) = \sum_{t=1}^{T} \mu_{m,t} e_{r,t} + p_{m,t} e_{s,t} + F_{m,t}, \tag{10}$$

where $E_t(s_t, a_t)$ denotes the system energy consumption at time t, and we give expectations for whole MTT-WSNs. $F_{m,t}$ is the fetching cost corresponding to the Equation (2).

When execution latency exceeds the given deadline, the delay penalty is generated and is denoted by

$$D(\pi) = \sum_{t=1}^{T} D_t(s_t, a_t), \tag{11}$$

$$D_t(s_t, a_t) = \sum_{n=1}^{N} x_n(I_{n,t}), \tag{12}$$

where $D_t(s_t, a_t)$ denotes the penalty of the system delay, and $x(*)$ denotes the delay of total target nodes with diverse delay constraints. In order to ensure real-time tracking, target nodes with different important degrees are detected and tracked, and the trajectories in dynamic scenarios are random. Therefore, our purpose is to minimum the delay and the penalty of the system delay that can be written as

$$D_t(s_t, a_t) = \begin{cases} x_n(\theta_{n,t}), & \text{if} \quad \theta_{n,t} \leq \bar{I}_n, \\ y_n(\theta_{n,t}), & \text{if} \quad \theta_{n,t} > \bar{I}_n \end{cases}, \tag{13}$$

where $\theta_{n,t}$ denotes the practice delay for tracking target n at time t. Any delay will be punished, and the punishment

intensity $y_n(*)$ is much weighted than $x_n(*)$. Once the delay $\theta_{n,t}$ exceeds the threshold $I_{n,t}$, the punishment will gradually increase.

For a dynamic MTT-WSNs, it is difficult to use polynomial time to obtain the optimal policy, so the deep Q-network algorithm is employed in the MTT-WSNs. Our object is to minimize the systematic energy consumption and the system delay jointly. Therefore, the utility function of the joint optimization can be described as

$$U(\pi) = \xi_1 E(\pi) + \xi_2 D(\pi), \tag{14}$$

where ξ_1 and ξ_2 are the weighted parameters which satisfy $\xi_1 + \xi_2 = 1$, respectively. At time t,

$$U(s_t, a_t) = \xi_1 E(s_t, a_t) + \xi_2 D(s_t, a_t), \tag{15}$$

denotes the utility function to minimize the system cost.

As mentioned above, an optimal strategy π that minimizes the system cost should be obtained. Consequently, the scheduling problem is described as

$$P1 : u^* = \min U(s_t, a_t), \tag{16}$$

where u^* is the optimal joint solution produced by optimal policy π^*. Then, the optimal strategy is learned through interaction between the agent and the MTT-WSNs.

4.2. DQN-Based Scheduling Strategy. The scheduling process is modelled as MDP with quaternary-state which includes state space, action space, penalty, and the state transformed equation. We use a value function based on the Bellman equation to define the potential value for state s at time t in the future which is given by

$$P2 : v_t(s) = \frac{1}{T} E\left\{ \sum_{t=1}^{T} \gamma^t U(s_t, a_t) | s_1 = 0 \right\}, \tag{17}$$

where γ is the discount factor, and $P2$ denotes the exception function for a long-term accumulation process. The iteration process is represented as follows

$$v_{t+1}(s) = \min \sum_{s', D_t} p\left(s', D_t | s, a\right) \left(D_t + \gamma v_{t+1}\left(s'\right)\right). \tag{18}$$

In DQN, the agent interacts with the dynamic environment using a dynamic programming method to alleviate the above issue. The learning process with Q value is represented as follows, which is embodied as an approximating value function,

$$Q(s_t, a_t) = E[D_t + E[Q(s_{t+1}, a_{t+1})]], \tag{19}$$

and the updating process is given by

$$Q(s, a) = Q(s, a) + \alpha \left[D + \max_{a'} \left(Q\left(s', a'\right) - Q(s, a) \right) \right],$$

(20)

where α is the learning rate.

In Figure 2, we describe the specific interaction process. In order to minimize the loss, state variables as input of primary network are learned and updated through replay memory. Each time the network is trained, a sample is selected randomly to reduce the relevance of data. To improve the stability of the algorithm, target network is established, the same as the primary network which is used to update the network parameters. Target network is applied to update Q value. It is noteworthy that the two networks are asynchronous.

Meanwhile, the DQN-based scheduling strategy algorithm is shown below,

The time complexity is given for algorithm analysis. Based on the mention above discussions, the feedback process of neural networks is mainly considered with high resource consumption. It implements the matrix computing with inverse operation. For our designed intelligent computing algorithm, we focus on the constructed hidden layers of neural networks and the number of iterations. Explicitly, the action a is sampled from a set of action $\mathscr{A}$ with a time complexity $\mathscr{O}(1)$ for each iteration [22]. In the hidden layers, the main consideration focuses on back propagation. According to [12], its time complexity is $O(k(v)TM_1)$. Thereafter, the algorithmic complexity is given as $\mathscr{O}(k(\phi)T\mathscr{M}_1|\mathscr{C}\|\mathscr{A}_1|)$, where $\mathscr{A}_1$ and $\mathscr{M}_1$ are the reduced action space and set of target nodes, $\mathscr{C}$ is the number of sink nodes, and $k(v)$ is the a function of the depth and number of the hidden layers v.

5. Performance Evaluation

In this section, we present numerical results to illustrate the performance of the proposed optimal policy.

5.1. Simulation Set-up. To begin with, the DQN agent obtains policy from MTT-WSNs' environment. Considering the mobile target tracking scenario in a stable region, we set the number of target nodes as $\mathscr{N} = 2$, the number of sensor nodes as $\mathscr{M} = 40$, and the distribution of sensor nodes is uniform. Meanwhile, the velocity of sensor nodes and targets is 2 m/s and 1 m/s, respectively. The maximum tolerate delays are 5 seconds and 6 seconds, respectively. We set the primary energy of each sensor node as 40 J. The coordinates of running into each region for two targets are given, and the moving trajectory is the stochastic distribution. The energy consumption of static nodes is set as 0.2 J and 0.8 J for each moving sensor. Parameters of the MTT-WSNs' environment in our system are listed in Table 1. The simulation is based on Python 3.0 and Tensor Flow framework.

6. Simulation Results and Discussion

Figure 3 depicts the systematic energy consumption under the consideration of fairness between energy and delay, i.e., $\xi_1 = \xi_2 = 0.5$. With the different learning rates, system consumption is converging quickly when the number of iterations arrives at 120 steps approximately. It is obvious that different learning rates give diverse capability for convergence, and convergent velocity is faster correspondingly when the learning rate is equal to 0.6 which has the least fluctuations and fastest convergence speed after 90 iterations. When the value of the horizontal ordinate is 90, the agent can use the learned experience to schedule diverse sensor nodes considering self-energy and relative distance from each target.

Under the consideration of collaborative perception, we also obtain the performance of trade-off between energy and delay penalty. When we mainly consider energy, i.e., $\xi_1 = 0.9, \xi_2 = 0.1$ in Figure 4, the performance of convergent velocity is also compared with Figure 3. In this case, the lower energy consumption is realized while guaranteeing the convergence. When systematic energy consumption is convergent, energy consumption is reduced to 42.85% approximately compared with Figure 3, wherein energy consumption and delay are analysed simultaneously.

Figure 5 depicts the systematic delay cost, which mainly considers the delay, i.e., $\xi_1 = 0.1, \xi_2 = 0.9$. Tendency is convergent when the value of the horizontal ordinate is 75 and the learning rate is 0.9. Due to the randomly sampled replay and updated memory, the convergence exists a little tremble, and the performance is better than convergence of systematic energy consumption.

Figure 6 shows the comparison with the Randomly Selecting Node (RSN) algorithm. Considering the energy and delay fairly, we obviously extract that the capability of DQN-based collaborative perception algorithm is better than RSN. We analyse the whole interacting process of the agent and MTT-WSNs when the value of horizontal ordinate is fewer than 120, and the energy consumption is reduced to 76.2%, 67.3%, and 70.9% when learning rate are 0.6, 0.8, and 0.9, respectively. The effect is obviously remarkable compared with RSN not only in energy consumption but also when the convergence outperforms RSN algorithm.

7. Conclusions

In this paper, we have studied the mobile multitarget tracking scheduling management in the complex MTT-WSNs. The problem of mobile scheduling is formulated as a MDP, taking account of diverse important degree for different target nodes and diverse tracking capabilities for different sensor nodes. Moreover, the multiregional tracking and scheduling strategy is treated as an important component to reduce the size of state space with a low computational complexity. To cope with the mobile multitarget tracking in a complex scenario, collaborative perception among sensor nodes and the deep Q-learning algorithm is adopted to obtain the optimal scheduling strategy. Since the agent learns important information from MTT-WSNs through

the interacting process, the proposed scheme can enhance the network performance obviously and reduce the tracking delay as illustrated in simulation experiments. Furthermore, the systematic energy consumption can be reduced by the scheduling policy through sink nodes. Finally, the time complexity is analysed to promote further work smoothly.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] K. Xiong, S. Leng, J. Hu, X. Chen, and K. Yang, "Smart network slicing for vehicular fog-RANs," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 4, pp. 3075–3085, 2019.

[2] E. Mazor, A. Averbuch, Y. Bar-Shalom, and J. Dayan, "Interacting multiple model methods in target tracking: a survey," *IEEE Transactions on Aerospace & Electronic Systems*, vol. 34, no. 1, pp. 103–123, 1998.

[3] C. H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 685–692, CA, USA, June 2010.

[4] C. Liang and Y. U. Shao-Hua, "Preliminary study on the trend of 6G mobile communication," *Study on Optical Communications*, vol. 45, no. 4, pp. 1–8, 2019.

[5] E. Calvanese Strinati, S. Barbarossa, J. L. Gonzalez-Jimenez, D. Kténas, N. Cassiau, and C. Dehos, "6G: the next frontier," 2019, http://arxiv.org/1901.

[6] M. Z. A. Bhuiyan, G. Wang, and A. V. Vasilakos, "Local area prediction-based mobile target tracking in wireless sensor networks," *IEEE Transactions on Computers*, vol. 64, no. 7, pp. 1968–1982, 2015.

[7] J. Luo, Z. Zhang, C. Liu, and H. Luo, "Reliable and cooperative target tracking based on WSN and WiFi in indoor wireless networks," *Access*, vol. 6, pp. 24846–24855, 2018.

[8] C. Fan, B. Li, J. Hou, Y. Wu, W. Guo, and C. Zhao, "Robust fuzzy-learning for par-tially overlapping channels allocation in UAV *Communication Networks*," *IEEE Transactions on Mobile Computing*, 2018, https://arxiv.org/pdf/1806.10756.

[9] T. Zhao, S. Zhou, X. Guo, and Z. Niu, "Tasks scheduling and resource allocation in heterogeneous cloud for delay-bounded mobile edge computing," *2017 IEEE International Conference on Communications*, 2017, pp. 1–7, Paris, France, May 2017.

[10] L. Wan, X. Wang, and G. A. Bi, "Range-Doppler-Angle *Estimation Method for Passive Bistatic Radar*," 2017.

[11] G. Qiao, S. Leng, Y. Zhang, and L. Xu, "Joint multi-RATs and cloud-service matching scheme in wireless heterogeneous networks," *IEEE International Conference on Communications*, 2016, pp. 1–6, Kuala Lumpur, Malaysia, May 2016.

[12] G. Qiao, S. Leng, M. Zeng, and Y. Zhang, "Matching game approach for charging scheduling in vehicle-to-grid networks," *ICC 2017 IEEE International Conference on Communications*, 2017, pp. 1–6, Paris, France, May 2017.

[13] S. J. Nawaz, S. K. Sharma, S. Wyne, M. N. Patwary, and M. Asaduzzaman, "Quantum machine learning for 6G communication networks: state-of-the-art and vision for the future," *IEEE Access*, vol. 7, pp. 46317–46350, 2019.

[14] C. Sergiou, M. Lestas, P. Antoniou, C. Liaskos, and A. Pitsillides, "Complex systems: a communication networks perspective towards 6G," *Access*, vol. 8, pp. 89007–89030, 2020.

[15] M. Z. Chowdhury, M. Shahjalal, S. Ahmed, and Y. M. Jang, "6G wireless communication systems: applications, requirements, technologies, challenges, and research directions," *IEEE Open Journal of the Communications Society*, vol. 1, pp. 957–975, 2020.

[16] K. Chia-Hsu, C. Tzung-Shi, and S. Siou-Ci, "Robust mechanism of trap coverage and target tracking in mobile sensor networks," *IEEE Internet of Things Journal*, vol. 5, no. 4, pp. 3019–3030, 2018.

[17] X. Wang, J. Guo, and N. Cui, "Adaptive extended Kalman filtering applied to low-cost MEMS IMU/GPS integration for UAV," *International Conference on Mechatronics and Automation*, vol. 2009, pp. 2214–2218, 2009.

[18] H. Zhang, X. Zhou, Z. Wang, H. Yan, and J. Sun, "Adaptive consensus-based distributed target tracking with dynamic cluster in sensor net-works," *IEEE Transactions on Cybernetics*, vol. 49, no. 5, pp. 1580–1591, 2018.

[19] J. Li, Z. Xing, W. Zhang, Y. Lin, and F. Shu, "Vehicle tracking in wireless sensor networks via deep reinforcement learning," *IEEE Sensors Letters*, vol. 4, no. 3, pp. 1–4, 2020.

[20] Z. Yuan, X. Huang, L. Sun, and J. Jin, "Software defined mobile sensor network for micro UAV swarm," *IEEE International Conference on Control and Robotics Engineering (ICCRE)*, vol. 2016, pp. 1–4, 2016.

[21] S. Mori, "Cooperative sensing data collecting framework by using unmanned aircraft vehicle in wireless sensor network," *IEEE International Conference on Communications (ICC)*, vol. 2016, pp. 1–6, 2016.

[22] L. Zhou, S. Leng, Q. Liu, S. Mao, and Y. Liao, "Intelligent resource collaboration in mobile target tracking oriented mission-critical sensor networks," *IEEE Access*, vol. 8, pp. 10971–10980, 2020.

[23] L. A. Prashanth, A. Chatterjee, and S. Bhatnagar, "Two time-scale convergent Q-learning for sleep-scheduling in wireless sensor networks," *Wireless Networks*, vol. 20, no. 8, pp. 2589–2604, 2014.

[24] P. Wu, J. Li, L. Shi, M. Ding, K. Cai, and F. Yang, "Dynamic content update for wireless edge caching via deep reinforcement learning," *IEEE Communications Letters*, vol. 23, no. 10, pp. 1773–1777, 2019.

[25] J. Wu and G. Li, "Drift calibration using constrained extreme learning machine and Kalman filter in clustered wireless sensor networks," *IEEE Access*, vol. 8, pp. 13078–13085, 2020.

[26] X. Bai, Z. Wang, L. Zou, and C. Cheng, "Target tracking for wireless localization systems with degraded measurements and quantization effects," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 12, pp. 9687–9697, 2018.

Journal of Sensors

[27] J. Gu, T. Su, Q. Wang, X. Du, and M. Guizani, "Multiple moving targets surveillance based on a cooperative network for multi-UAV," *IEEE Communications Magazine*, vol. 56, no. 4, pp. 82–89, 2018.

[28] L. Zhou, S. Leng, Q. Liu, and Q. Wang, "Intelligent UAV swarm cooperation for multiple targets tracking," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 743–754, 2022.

Research Article

Adaptive Bright and Dark Channel Combined with Defogging Algorithm Based on Depth of Field

Shiqi Wang ⓘ, Tingping Yang ⓘ, Wenxue Sun ⓘ, Xiao Lu ⓘ, and Di Fan ⓘ

Shandong University of Science and Technology, Qingdao, 266590 Shandong, China

Correspondence should be addressed to Di Fan; fandi_93@126.com

Aiming at the problem that the dark channel prior algorithm is not ideal in removing fog at the distant view and the sudden change of depth, this paper proposes an adaptive bright and dark channel combined with defogging algorithm based on the depth of field. The algorithm divides the image into a distant area and a near area based on the depth of field. The near area uses the dark channel to remove the fog, and the distant area uses the bright channel to remove the fog, combined with the use of adaptive superpixel segmentation to give the content-based local area division and defogging control factor based on depth of field. It better solves the problem of large transmittance deviation and defogging distortion under content changes and sudden depth of field changes. From the subjective and objective indicators, the algorithm in this paper has achieved a good defogging effect, and the main indicators and comprehensive defogging quality are better than common algorithms.

1. Introduction

Fog has a great impact on the application of visual recognition in intelligent transportation. Dense fog will lead to system imaging blur and recognition failure. Images taken by image sensors in poor weather environments such as fog, rain, and haze will have serious degradation problems, which brings many difficulties in extracting useful information from images and has an important impact on the application of remote sensing, target detection, intelligent transportation, and other fields [1, 2]. The research of image defogging has received widespread attention. The existing image defogging algorithms are mainly divided into enhancement algorithms, restoration algorithms, and deep learning algorithms.

The enhanced defogging algorithm improves the image quality through image enhancement technology, mainly including adaptive histogram equalization [3, 4], wavelet transform [5, 6], homomorphic filtering [7], and Retinex enhancement [8–10] algorithms. The adaptive histogram equalization defogging algorithm [3, 4] is an improvement on the basic histogram algorithm, which can indistinguishably improve the image contrast, suppress the slope of the

transformation function to some extent, and avoid the phenomenon that rising too fast resulting weak image contrast and oversaturation. However, such methods will amplify the noise in the image when there is a lot of noise in the image. The wavelet transform method divides the image into high-frequency region and low-frequency region and uses the enhancement method for the high-frequency region to achieve the purpose of image defogging by improving the image contrast [5, 6], but it is not suitable for the situation of too bright or too dark and uneven illumination. The homomorphic filtering algorithm composes the illumination component and reflection component of the image, respectively, and processes them in the frequency domain, highlighting the details by enhancing the high-frequency information of the image [7]. It can effectively solve the problem of uneven illumination, but the Fourier transform used causes high computational complexity. The defogging algorithm based on Retinex generally adopts the multiscale Retinex with color restoration (MSRCR) method, which can obtain good defogging effect to a certain extent, but the defogging effect in dense fog scene is not ideal [8–10]. Thanh et al. successively proposed single image dehazing based on adaptive histogram equalization and linearization of gamma correction [11] and

single image dehazing with optimal color channels and nonlinear transformation [12]. The method is fast and effective, and the processed image is better than the comparison algorithm in visual and objective indexes.

Restoration defogging algorithms mainly include defogging algorithms based on partial differential equations [13, 14] and defogging algorithms based on prior knowledge [15–25]. The defogging algorithm based on partial differential equation uses multiple images to realize image defogging according to the polarization characteristics of light. Wu et al. [13] and Guo and Meng [14], respectively, proposed a series of improvement measures for the partial differential equation model. The defogging algorithm based on prior knowledge infers the cause of image degradation based on assumptions or prior information and estimates the atmospheric light and transmittance based on the atmospheric scattering model to obtain a fog-free image. The most typical one is the dark channel prior defogging algorithm. The dark channel prior defogging algorithm proposed by He et al. [15] has better effects in close and dark areas, but there are problems with distortion and poor defogging effects in distant and bright areas. Yan et al. proposed the bright channel prior [16], which uses the combination of bright channel and dark channel prior to achieve image deblurring, but did not consider the influence of different depths of field on the dehazing effect. Gao et al. proposed a far and near scene fusion defogging algorithm based on the prior of dark-light channel, and the saturation and sharpness of the image have been improved to a certain extent [17]. Yang et al. proposed adaptive haze estimation and transmittance estimation algorithms and achieved a certain dehazing effect [18, 19]. However, there is oversaturation in areas where dense fog and mist change drastically. Zhu et al. proposed a color attenuation prior. The depth of field information is obtained based on this prior, but the sample collection process of this method is difficult, and the theoretical basis is slightly lacking [20]. Fan et al. successively proposed an adaptive defogging algorithm based on color attenuation [21], single image defogging algorithm based on three-region division [22], an iterative defogging algorithm based on pixel-level atmospheric light map [23], and image defogging algorithm based on sparse representation [24]. The sharpness and contrast of the image after defogging are greatly improved. Kumar et al. proposed a region-based adaptive denoising and detail enhancement method, proposed the autocolor transfer method to strengthen the dark regions of the hazy image, and also considered the variation in haze levels in different regions of the image for adaptive adjustment [25].

The deep learning defogging algorithm achieves defogging by training the model. It has a good defogging effect, but it requires a lot of calculation and needs to build a large dataset [26]. Cai et al. proposed to use the DehazeNet convolutional neural network to estimate the transmittance of the fog map, but the convolutional neural network takes a long time and has poor real-time performance [27]. Luan et al. proposed a classification algorithm based on a learning framework, using a support vector regression model to obtain accurate transmittance, but the feature extraction process is complicated [28]. Liu et al. designed an end-to-end convolutional neural network GridDehazeNet for image defogging and realized multiscale estimation based on attention [29]. Such algorithms require a large amount of datasets during the training process and are prone to overfitting. It is suitable for a single scene and depends too much on the foggy images and nonfoggy images in the datasets. The defogging effect in the real scene is poor.

Aiming at the deficiencies of the above research, this paper proposes an adaptive bright and dark channel combined with defogging algorithm based on depth of field, which solves the problem of distortion in the distant view and poor defogging effect in the sudden depth of field. The segmentation window is adaptively determined based on the image content, and the defogging control factor and transmittance are adaptively determined based on the depth of field. The dark channel is used to defog in near areas and dark areas, and the bright channel is used to defog in distant areas and bright areas, so as to give play to their advantages in different areas. At the same time, based on the depth of field, the dark channel and the bright channel are adaptively divided and the defogging control factor is determined; the superpixel segmentation algorithm is used to give the content-based division of the local areas in the two channels. The two results are applied to the bright and dark channel defogging algorithm, which solves the problem in the original dark channel defogging algorithm. From the experimental results, good defogging effects have been obtained in both the distant and near areas, and the main indicators and comprehensive defogging quality are better than common algorithms.

2. Atmospheric Scattering Model and Principle of Bright and Dark Channels

2.1. Atmospheric Scattering Model. The atmospheric scattering model describes the imaging process under fog and haze conditions, and it is the theoretical basis of the restoration image defogging algorithms. Its mathematical model is

$$I(x) = J(x) * t(x) + A * (1 - t(x)). \tag{1}$$

In the formula, x represents the image pixel position, $I(x)$ represents a foggy image, $J(x)$ represents a fog-free image, $t(x)$ represents transmittance, and A represents the atmospheric light. If β represents the atmospheric scattering coefficient and $d(x)$ represents the depth of field, that is, the distance from the target position to the camera, the transmittance $t(x)$ can be expressed as

$$t(x) = e^{-\beta d(x)}. \tag{2}$$

According to the atmospheric scattering model, when $t(x)$ and A are obtained, the fog-free image $J(x)$ can be obtained backward according to formula (1).

2.2. Dark Channel Prior. The dark channel prior is the rule that He et al. [15] summarized by counting the characteristics of a large number of outdoor fog-free images, that is,

for a fog-free image, in addition to the sky and bright areas, the value of pixels in its local area at least on one channel is very low and tends to zero. For any input image, its dark channel can be defined as

$$J^{\text{dark}}(x) = \min_{y \in \Omega(x)} \left(\min_{c \in \{r,g,b\}} J^c(y) \right), \longrightarrow 0 \qquad (3)$$

where min represents the operation of taking the minimum value, $\Omega(x)$ represents the local area centered on pixel x, c represents one of the three channels of r, g, b, $J^c(y)$ represents the pixel in area $\Omega(x)$, and $J^{\text{dark}}(x)$ represents the dark channel image.

Assuming that the atmospheric light value A is known, and in the window $\Omega(x)$, the transmittance $t(x)$ is a fixed value and denoted as $\tilde{t}(x)$. The minimum values of both sides of equation (1) are

$$\min_{y \in \Omega(x)} \left(\min_c I^c(y) \right) = \tilde{t}(x) * \min_{y \in \Omega(x)} \left(\min_c J^c(y) \right) + A * \left(1 - \tilde{t}(x)\right). \qquad (4)$$

The dark channel value of a foggy image is represented by $I^{\text{dark}}(x)$, then

$$I^{\text{dark}}(x) = \tilde{t}(x) * J^{\text{dark}}(x) + A * \left(1 - \tilde{t}(x)\right). \qquad (5)$$

It can be known from the dark channel prior that $J^{\text{dark}}(x)$ approaches 0, and the above formula can be approximated as

$$I^{\text{dark}}(x) = A * \left(1 - \tilde{t}(x)\right). \qquad (6)$$

The transmittance can be obtained by formula (6):

$$\tilde{t}(x) = 1 - w(x) * \frac{I^{\text{dark}}(x)}{A}, \qquad (7)$$

where $w(x)$ is the defogging control factor, which is used to adjust the defogging intensity and prevent the image distortion caused by too strong defogging. $w(x)$ can take a fixed value (such as $w = 0.95$) or a variable value.

Dark channel defogging is the process of obtaining the transmittance $\tilde{t}(x)$ according to the foggy image $I(x)$ and the estimated atmospheric light A and then obtaining the defogging image according to the atmospheric scattering model.

2.3. Bright Channel Prior. According to the dark channel prior, the dark channel values of sky area, dense fog area, bright area, and white area are relatively large, which does not satisfy the dark channel prior. Yan et al. [16] proposed the bright channel prior through statistical analysis, that is, in natural scene images, for image blocks of white objects, sky, bright areas, light intensity areas, etc., there is at least one color channel with a very large value, close to 1. The bright channel of an image can be defined as

$$J^{\text{bright}}(x) = \max_{y \in \Omega(x)} \left(\max_{c \in \{r,g,b\}} J^c(y) \right), \longrightarrow 1 \qquad (8)$$

where $J^{\text{bright}}(x)$ represents the bright channel image, max represents the operation of taking the maximum value.

Similar to the dark channel prior defogging process, assuming that the atmospheric light value A is known, and the transmittance $t(x)$ in the window $\Omega(x)$ is a fixed value $\tilde{t}(x)$. The maximum values on both sides of equation (1) are

$$\max_{y \in \Omega(x)} \left(\max_c I^c(y) \right) = \tilde{t}(x) * \max_{y \in \Omega(x)} \left(\max_c J^c(y) \right) + A * \left(1 - \tilde{t}(x)\right). \qquad (9)$$

The left side of the above formula is the bright channel of the foggy image, denoted as $I^{\text{bright}}(x)$; then,

$$I^{\text{bright}}(x) = \tilde{t}(x) * J^{\text{bright}}(x) + A * \left(1 - \tilde{t}(x)\right). \qquad (10)$$

According to the theory of bright channel prior, $J^{\text{bright}}(x)$ approaches 1, which can be approximately obtained as follows:

$$I^{\text{bright}}(x) = \tilde{t}(x) + A * \left(1 - \tilde{t}(x)\right). \qquad (11)$$

By simplifying the above formula and introducing the defogging control factor $w(x)$, the transmittance can be obtained:

$$\tilde{t}(x) = 1 - w(x) * \frac{\left(1 - I^{\text{bright}}(x)\right)}{(1 - A)}. \qquad (12)$$

Similarly, a certain degree of defogging can be achieved according to the bright channel prior.

Obviously, dark channel defogging and bright channel defogging have their own applicable areas and failure areas, so both have their limitations. However, after analysis, it is not difficult to find that the applicable areas of them are complementary. Therefore, this paper proposes an adaptive bright and dark channel combined with defogging algorithm based on depth of field, which combines the two defogging methods. For the area where the dark channel prior is applicable, the dark channel is used to obtain the transmittance. For the area where the bright channel prior is applicable, the bright channel is used to obtain the transmittance. The areas are adaptively determined by the depth of field, and the defogging control factor also changes adaptively following the depth of field. The window $\Omega(x)$ is a content-based special-shaped area obtained by the superpixel segmentation method.

3. Adaptive Bright and Dark Channel Combined with Dehazing Algorithm Based on Depth of Field

The framework of adaptive bright and dark channels combined with defogging algorithm based on depth of field is shown in Figure 1. The algorithm is mainly divided into
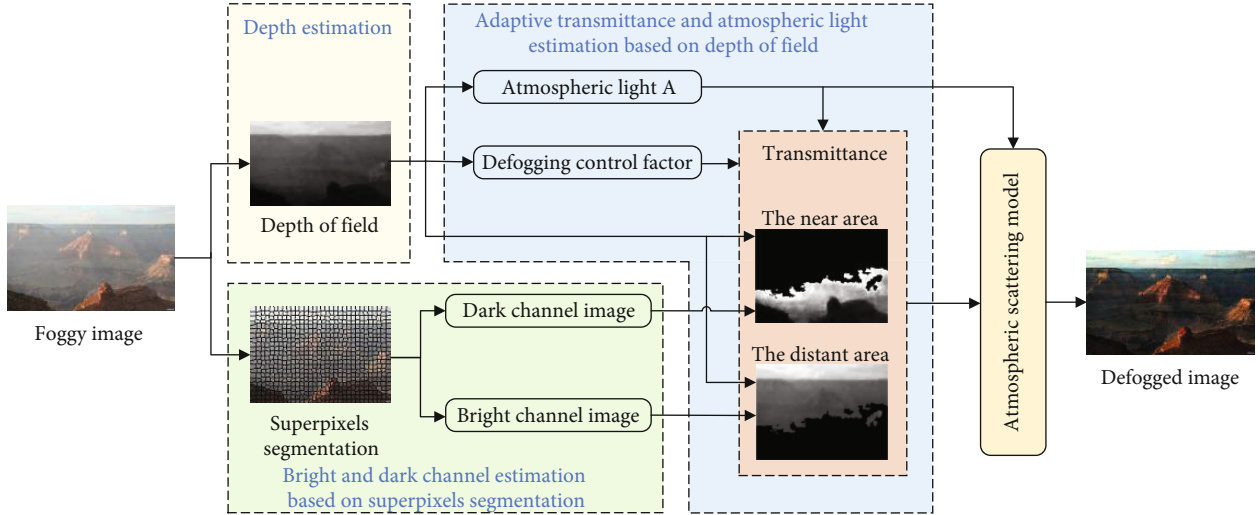
FIGURE 1: The framework of adaptive bright and dark channels combined with defogging algorithm based on depth of field.

three parts: depth estimation, adaptive transmittance, and atmospheric light estimation based on depth of field, and bright and dark channel estimation based on superpixels. According to the positive correlation between depth of field and brightness [18], the Retinex method was used to calculate brightness and then obtain the estimation of depth of field. The maximum depth of field is the sky, which is approximate to atmospheric light, so atmospheric light A is estimated from the maximum depth of field. At the same time, an adaptive defogging control factor $w(x)$ is obtained according to the negative correlation between the defogging control factor and the depth of field. According to the depth of field, the near area and the distant area are separated. The dark channel was used to obtain the transmittance in the near areas, and the bright channel was used to obtain the transmittance in the distant area. The local area in the bright and dark channel is obtained by adaptive superpixel segmentation method. The obtained area changes according to the content, and the shape is irregular, which is more in line with the characteristics of the image, and solves the problem that the content jump cannot be adapted when the regular area is divided. After obtaining atmospheric light and transmittance, a defogging image is obtained according to the atmospheric scattering model.

3.1. Depth Estimation Algorithm Based on Retinex. Observing statistics on multiple foggy images, it is found that the fog density in the foggy image increases with the increase of the scene depth; the brightness of the foggy image gradually increases with the increase of the fog density [18]. Therefore, it can be assumed that the depth of field is positively correlated with the fog density and also positively correlated with the image brightness, that is,

$$d(x) \propto c(x) \propto v(x), \tag{13}$$

where $d(x)$, $c(x)$, and $v(x)$, respectively, represent the scene depth, fog density, and brightness value at x, and '$\propto$' represents a direct ratio.

According to the characteristics of the positive correlation between the depth of field brightness, this paper first uses Retinex to obtain the fog mask [30] and then obtains the optimized brightness map, so as to obtain the image depth of field. The specific process is shown in Figure 2.

First, the foggy image is transferred to the YCbCr color space, the normalized brightness component $Y(x)$ is extracted, and the Gaussian smoothing function $F(x)$ is used to convolve with it to obtain the image fog mask $\widehat{L}(x)$, that is,

$$\widehat{L}(x) = Y(x) \otimes F(x),$$
$$F(x) = K * e, \tag{14}$$

where '$\otimes$' represents the convolution operation; K is the normalization factor; σ is the standard deviation, the magnitude of which controls the smoothness of the function $F(x)$; m and n represent the horizontal and vertical coordinates at x.

The uniform fog mask $\bar{L}$ is the mean value of $\widehat{L}(x)$, namely,,

$$\bar{L} = \frac{\sum_{m=1}^{H}\sum_{n=1}^{W}\widehat{L}(x)}{H * W}, \tag{15}$$

where H and W represent the height and width of the image, respectively. However, $\bar{L}$ is only applicable when the fog is evenly distributed. In reality, the fog distribution is positively correlated with the depth of field. Therefore, in this paper, the uniform fog mask $\bar{L}$ and the brightness component $Y(x)$ of the foggy image are processed according to equation (16) to obtain the fog mask $L'(x)$ related to the depth of field:

$$L'(x) = \text{Guided Filter}\left(Y(x), \left(1 - Y(x) * \bar{L}\right), r, \text{eps}\right), \tag{16}$$

where Guided Filter represents the guided filtering operation, $Y(x)$ represents the guided image, $(1 - Y(x) * \bar{L})$ represents
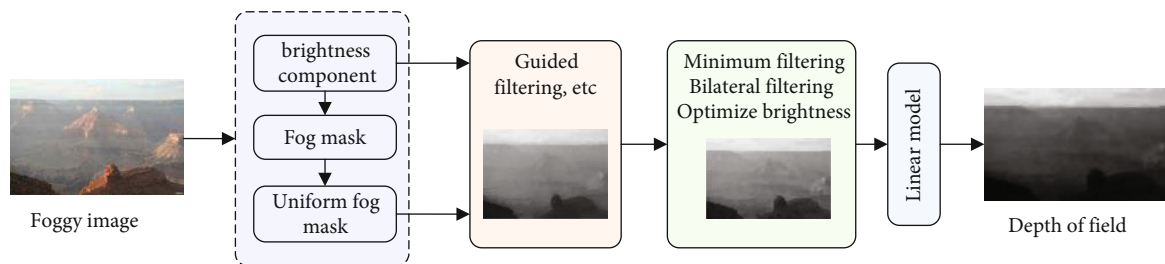
FIGURE 2: Depth estimation algorithm framework.

the input image, r represents the local window radius, and eps represents the regularization parameter.

Since high-brightness areas such as white objects will be misjudged as those with a far depth of field, a minimum filter of 9×9 and bilateral filtering is performed on $L'(x)$ to eliminate blocking effects and optimize to obtain the brightness map $L(x)$; Finally, according to the linear relationship model between the depth of field and the brightness component [16], the depth map $d(x)$ is obtained:

$$d(\mathrm{x}) = \frac{L(x) - L_{\min}}{L_{\max} - L_{\min}} * L(x), \qquad (17)$$

where $L_{\min}$ and $L_{\max}$ represent the minimum and maximum values of the optimized brightness map $L(x)$, respectively.

The depth of field obtained based on the Retinex depth of field estimation algorithm is shown in Figure 3.

3.2. Adaptive SLIC Superpixel Segmentation Algorithm. Superpixel segmentation [31, 32] divides adjacent and similar parts into a region, and the result is an irregular window which changes according to the content. The bright and dark channels and their transmittances are obtained by using the divided window instead of the fixed rectangular window, which can solve the block effect problem at the depth change well.

The size of the superpixel segmentation window has a great impact on the bright and dark channels and defogging effect. For the dark channel image, when the segmentation window is too large, there are more dark channels in the window, the high pixel value in the dark channel image is reduced, the details of the defogging image are blurred, the level is not obvious, and the edge of the dark channel image is easily expanded; If the segmentation window is too small, although the image details are clear and the levels are rich, the local bright area does not conform to the dark channel prior theory, and the pixel value is prone to be too high, resulting in inaccurate estimated transmittance. For the bright channel image, the opposite is true. In order to avoid the influence of different segmentation windows on the bright and dark channel images, this paper does not select a fixed segmentation window when obtaining the bright and dark channel images but adaptively determines the window size according to the size of the image. When the image resolution is relatively large, the segmentation window is correspondingly increased, and when the image resolution

is relatively small, the segmentation window is correspondingly reduced. This paper uses a simple linear iterative clustering algorithm (SLIC) for segmentation. The distance between the seed points of the segmented superpixel block is related to the parameter s. The larger the s, the larger the segmentation window, and vice versa. In this paper, the superpixel segmentation window size is controlled by determining the parameter s, which is as follows:

$$s = \mathrm{floor}\left(\max\left([15, H * 0.01, W * 0.01]\right)\right), \qquad (18)$$

where floor represents rounding towards negative infinity.

Figure 4 shows the segmentation results of the adaptive SLIC superpixel and the bright and dark channel images. Figure 4(a) is the result of super pixel segmentation of the image in Figure 3, from which it can be seen that the boundary of the super pixel block formed after segmentation is relatively clear. Figure 4(b) is the process image, the superpixel segmentation result is overlaid on the minimum channel image. Figure 4(c) is a dark channel image obtained by using the result of superpixel segmentation. The level is clear and the window size is moderate. In the contour of the mountain and the boundary area between the sky and the scene, the pixels with similar depth values can be better segmented into the same window. Figure 4(d) is the bright channel image obtained by using the superpixel segmentation result. It can be seen from the figure that in the sky and areas with relatively bright brightness, the pixel value is relatively high, and the bright area can be clearly seen in the image after segmentation.

3.3. Acquisition of Adaptive Transmittance. It can be seen from the foregoing that the idea of obtaining the transmittance is to first determine the defogging control factor $w(x)$ and atmospheric light A; Then, it is divided into the near area and the distant area, and the transmittance of each part is calculated according to equations (7) and (12) with dark channel and bright channel, respectively. After fusion, it becomes the transmittance of the image.

3.3.1. Determination of Adaptive Defogging Control Factor. The defogging control factor represents the degree of defogging. The defogging control factor in the dark channel defogging algorithm of He defaults to 0.95. This value has a better defogging effect for foggy images in the near area, while excessive defogging is likely to cause distortion in the distant area. Through experimentation, it is found that when
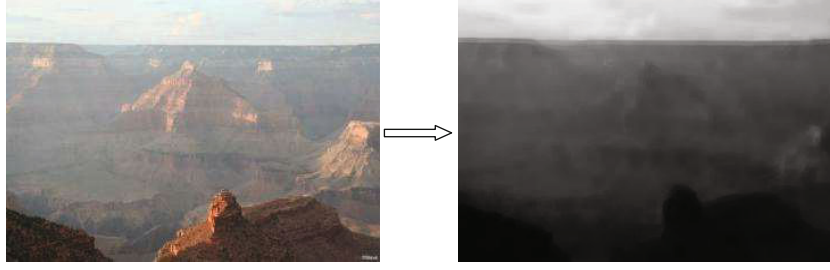
FIGURE 3: The original image and its depth of field map.

(a)

(b)

(c)

(d)
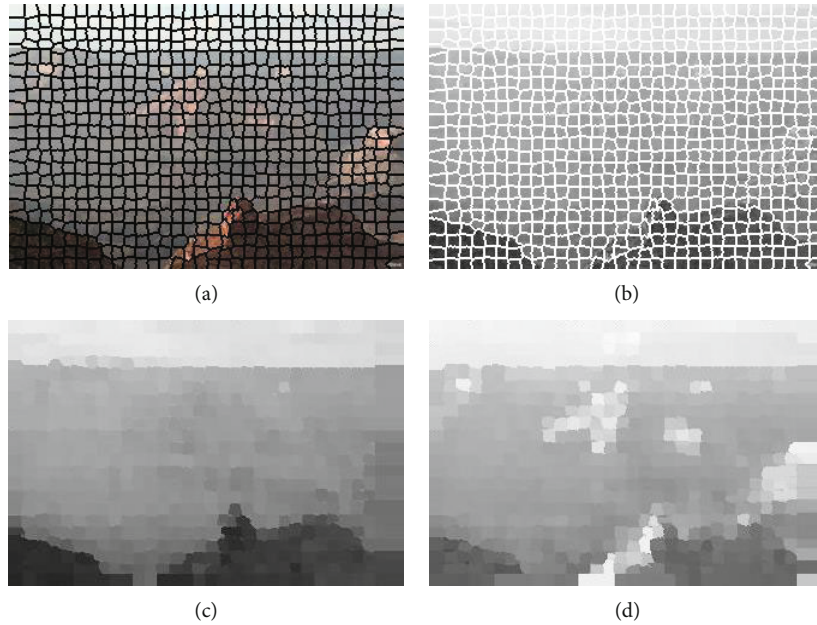
FIGURE 4: Adaptive SLIC superpixel segmentation results and bright and dark channel images, (a) original image adaptive SLIC superpixel segmentation, (b) adaptive SLIC superpixel segmentation with minimum channel image, (c) dark channel image, and (d) bright channel image.

the value is 0.55, the distortion problem at the distant area is solved, but the defogging effect at the near area is not ideal. In this paper, the nearest point is set to 0.95, and the farthest point is set to 0.55. According to the linear relationship that the depths of field $d(x)$ and w are negatively correlated, its size changes adaptively according to the depth of field value. The adaptive defogging control factor is obtained as follows:

$$w(x) = 0.95 - \frac{(0.4 * (d(x) - d_{\min}))}{(d_{\max} - d_{\min})}, \qquad (19)$$

where $d_{\min}$ and $d_{\max}$ represent the minimum and maximum values of the depth of field map, respectively.

3.3.2. Estimation of Atmospheric Light Value. The atmospheric light value is an important parameter in the physical model. The larger the atmospheric light value, the darker the restored image, and vice versa [33]. The depth of field of the image has been estimated in Section 3.1, the bright area represents the distant area, the brightness of the sky at infinity is similar to atmospheric light. Therefore, this

paper selects the top 0.1% of the brightest pixels in the depth map and uses the average value of the brightness of the foggy image pixels corresponding to these brightest pixels as the atmospheric light A.

3.3.3. Calculate the Transmittance of Bright and Dark Element Priors. According to the positive correlation between the depth of field and the fog density, the fog density in the distant view in the image is relatively large. The dense fog, highlights, and white areas in the image do not accord with the prerequisites for dark primary colors. This paper proposes an improved algorithm for bright and dark element priors. The dark channel is used to estimate the transmittance in the near area, and the bright channel is used to estimate the transmittance in the distant area. The division of the distant area and the near area is adaptively determined according to the depth of field. According to the characteristics of the depth of field, set the depth of field threshold d_{BL} to divide the bright channel and the dark channel. The area greater than the depth of field threshold is defined as the bright primary color area in the distant area,

(a)

(b)

(c)

(d)
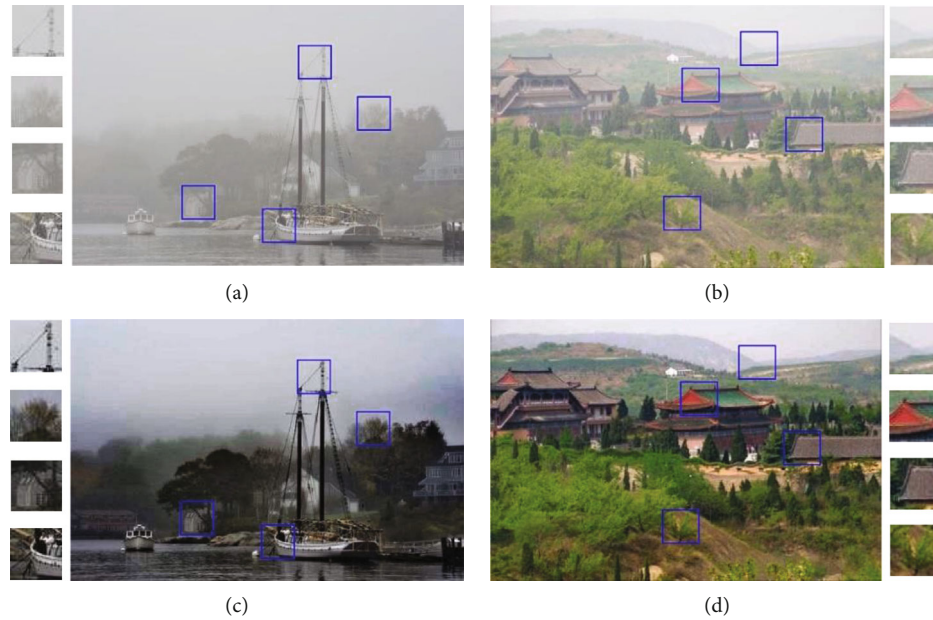
FIGURE 5: Comparison of the algorithm before and after defogging, (a) the first foggy image, (b) the second foggy image, (c) the first image after defogging, (d) the second image after defogging.

and the area less than the depth of field threshold is defined as the dark primary color area of the near area. And then the transmittance is obtained according to the dark and bright element priors:

$$
t(x) = \begin{cases} 1 - w(x) * I^{dark}(x)/A & d_{\min} \le d \le d_{BL} \\ 1 - w(x) * \left(1 - I^{bright}(x)\right)/(1-A) & d_{BL} \le d \le d_{\max} \end{cases}
$$

(20)

where $d_{BL} = d_{\min} + 3 * (d_{\max} - d_{\min})/4$. The parameter 3/4 in the threshold is the optimal value selected after multiple experiments. In order to ensure that the transmittance value is valid, the lower threshold value $t_0 = 0.1$ is set to limit the transmittance range, and the obtained atmospheric light A and transmittance $t(x)$ are substituted into equation (21) to obtain the defogging image $J(x)$:

$$
J(x) = \frac{(I(x) - A)}{\max \ (t(x), t_0)} + A
$$

(21)

4. Experimental Results and Analysis
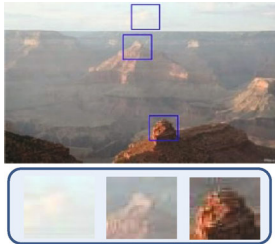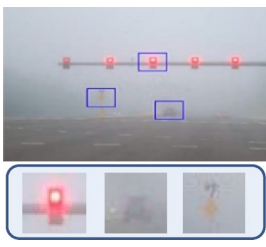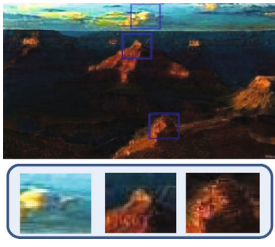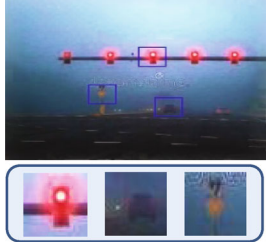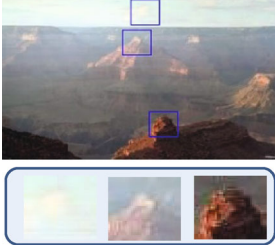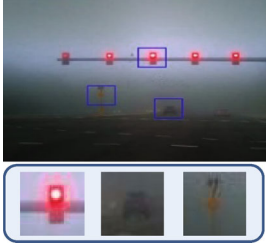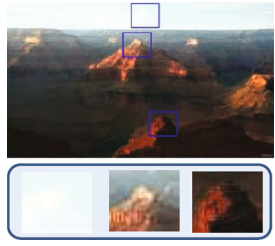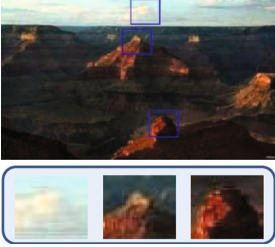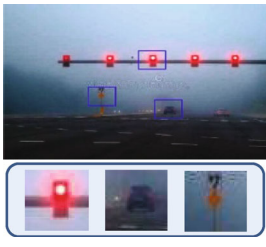
The experimental samples are selected from 5 foggy images in different scenes. The images contain different buildings, sceneries, sky, trees, etc., which are well representative. They are foggy images without ground truth. In order to better obtain experimental results, outdoor and indoor image pairs are added, respectively, for extended experiments, and O-HAZE [34] and I-HAZE [35] datasets are selected. O-HAZE contains 45 pairs of foggy images and corresponding fog-free outdoor images. I-HAZE contains 35 pairs of foggy images and corresponding fog-free indoor images. In this paper, we only use the foggy images in the datasets. It is

analyzed from both subjective and objective aspects and compared with He's algorithm [15], MSR algorithm [8], and three-region algorithm [22]. The experimental test environment is the MATLAB 2018a software platform, the CPU is Intel Core i5, the memory is 8GB, and the operating system is Windows 10.

4.1. The Experimental Results and Analysis of the Algorithm in This Paper. In this paper, several images containing sky regions and different sceneries are selected for experiments, and the results are shown in Figure 5. It can be seen from the figure that the contrast and sharpness of the image after defogging have been greatly improved, the overall defogging effect is better, and the distortion problem in the distant area has also been solved. Comparing the local magnification of the image before and after defogging, it can be seen that the image details are clear, the color is bright, and there is no too dark phenomenon. In Figure 5(c), the first image after defogging improves the contrast of the image as a whole. The details of the ship restored in the near view are richer, the outlines of houses and trees in the distant view are clearly visible, and the defogging effect of the image at the junction with the sky is good. In Figure 5(d), the second image after defogging has no color distortion and disorder as a whole. There is no oversaturation, and the color of the image is restored. Trees, roofs, etc. in the close range can better restore the sharpness of the image, avoiding the problem of overdarkness. The outline of the mountain in the distant view is also clearer.

4.2. Comparison with Other Algorithms. This paper selects three images with different sceneries and, respectively, uses the algorithm in this paper, He dark channel defogging algorithm [15], MSR algorithm [8], and three-area defogging algorithm [22] to conduct defogging experiments. The

TABLE 1: Experimental results of several defogging algorithms.

	Image 1s	Image 2	Image 3
Original image			
He algorithm			
MSR algorithm			
Three-area algorithm			
This paper algorithm			

results are shown in Table 1. The box below the figure is an enlarged view of a local area.

In image 1, the color distortion of the image processed by the He algorithm is serious, and the mountain area is too dark. Although there is no distortion after processing by the MSR algorithm, the details of the image after defogging are blurred and the definition is not high. After the three-area algorithm is processed, the overall defogging effect is good, but the cloud texture in the sky is not clear. The color of the image processed by the algorithm in this paper is not oversaturated, and the details are richer.

In image 2, after the He algorithm is processed, the image in the near view is too dark, and halos appear in some areas of the sky and red light areas. After the MSR algorithm is processed, the whole image has a color cast, the boundary between the sky and the nonsky area is not obvious, and the local details are not clear enough, which affects the visual effect. The color of the sky area processed by the three-area algorithm is unnatural, and a halo appears in the boundary area of the sky. The overall layering of the image processed by the algorithm in this paper is more distinct, the color is real and natural, and the overall brightness is moderate.

TABLE 2: Defogging evaluation indexes of several defogging algorithms.

Image	Defogging algorithms	PSNR	Mean Gradient	Information entropy	SSIM	e	NIQE	BRISQUE
Image 1	He	10.1357	0.0263	6.8766	0.5862	1.7784	3.4024	18.8705
	MSR	15.0943	0.0194	7.1412	0.6800	0.1615	3.4159	28.6994
	Three-area	10.2196	0.0312	7.0154	0.5846	1.0736	3.4442	29.5627
	This paper	12.8986	0.0244	7.2291	0.7349	2.2603	4.2828	28.7157
Image 2	He	12.1694	0.0144	7.5352	0.7645	2.1924	3.9991	42.0466
	MSR	11.2837	0.0072	7.3140	0.7863	1.2204	6.1299	45.8880
	Three-area	12.3719	0.0109	7.3370	0.7745	1.1582	5.3956	41.7019
	This paper	12.9303	0.0137	7.2816	0.8188	2.8474	4.4764	41.9678
Image 3	He	9.6292	0.0934	7.0651	0.6339	0.2011	5.6333	30.4511
	MSR	12.4145	0.0704	7.2044	0.9188	0.1285	4.1773	28.3339
	Three-area	10.9292	0.0698	7.5335	0.6673	0.1665	4.2227	20.0357
	This paper	13.0165	0.0986	7.7875	0.8102	0.2387	3.7285	20.9386
Average of three images	He	10.6448	0.0447	7.1590	0.6615	1.3906	4.3449	30.4561
	MSR	12.9308	0.0323	7.2198	0.7950	0.5035	4.5744	34.3071
	Three-area	11.1736	0.0373	7.2953	0.6755	0.7994	4.3542	30.4334
	This paper	12.9485	0.0456	7.4327	0.7880	1.7821	4.1626	30.5407

In image 3, the overall brightness of the image processed by the He algorithm is low, and the sky area is oversaturated, and the color is severely distorted. The overall color of the image processed by the MSR algorithm is true and natural, but the details are blurred. After the three-area algorithm processing, the sky area in the distant view is too bright, rough, and grainy. After processing by the algorithm in this paper, there is no overexposure in the distant view, and the color contrast of the image in the near view is maintained, the details are clear, and there is no overdark phenomenon.

In order to better evaluate the quality of defogging image, Table 2 lists the seven evaluation indicators of the experimental results in Table 1: Peak Signal to Noise Ratio (PSNR), Mean Gradient, Information Entropy, Structural Similarity (SSIM), ratio of newly added visible edges e, Natural Image Quality Evaluator (NIQE), and Blind Reference-less Image Spatial Quality Evaluator (BRISQUE). Among them, PSNR, SSIM, and e are reference image quality assessments; and Mean Gradient, Information Entropy, NIQE, and BRISQUE are blind image quality assessments. PSNR reflects the degree of image distortion. The larger the PSNR, the smaller the image distortion. The mean gradient reflects the ability of the image to express small details. The larger the mean gradient, the clearer the image. Information entropy measures the richness of image information and reflects the amount of information carried by the image. The greater the information entropy, the richer the information of the image and the better the quality of the image. e refers to the ratio of the newly added visible edges of the defogged image to the original foggy image. The larger the e is, the richer the edge details on the image and the better the contrast after defogging. NIQE can measure the difference of the image in the multivariate Gaussian distribution, and the smaller the value, the higher the quality of the image.

BRISQUE is a reference-free spatial domain image quality evaluation algorithm, which is evaluated based on the statistical law of image brightness. The smaller the result, the smaller the image distortion.

It can be seen from Table 2 that the PSNR of the three defogging images recovered by the algorithm is close to 13 on average, the mean gradient is 0.0456 on average, the information entropy is 7.4327 on average, the SSIM is 0.7880 on average, the ratio of newly added visible edges is 1.7821 on average, the NIQE is 4.1626 on average, and the BRISQUE is 30.5407 on average. Compared with the He algorithm, MSR algorithm and the three-region algorithm, the algorithm in this paper has the highest average value of PSNR, mean gradient, information entropy, and the ratio of newly added visible edges. The difference between SSIM, NIQE, and BRISQUE is not much different from the comparison algorithm. The results of all indexes show the effectiveness of the algorithm in this paper.

For better experimental analysis, we conducted extended experiments on O-HAZE and I-HAZE datasets. As shown in Table 3, the PSNR of the algorithm in this paper is close to 13.3 on average, the mean gradient is 0.019, the information entropy is 7.0301 on average, the SSIM is 0.7238 on average, the ratio of new visible edges is 2.7738 on average, the NIQE is 3.3171 on average, and the BRISQUE is 32.4422 on average. Compared with the other three algorithms, the values of PSNR, information entropy, and SSIM of the proposed algorithm are higher than those of He's algorithm and three-region algorithm, and slightly lower than those of MSR algorithm. The mean gradient is significantly higher than other comparison algorithms. The ratio of new visible edges is not different from that of the three-region algorithm, but obviously higher than that of the other two algorithms. The values of NIQE and BRISQUE are similar

TABLE 3: Defogging evaluation indexes on the O-Haze and I-Haze datasets.

Defogging algorithms	Datasets	PSNR	Mean Gradient	Information entropy	SSIM	e	NIQE	BRISQUE
He	O-HAZE	13.7448	0.0201	6.7726	0.7780	3.2390	3.1486	28.9011
	I-HAZE	11.2701	0.0133	6.7253	0.6674	1.5515	3.3470	37.9825
	Average	12.5075	0.0167	6.7490	0.7227	2.3953	3.2478	33.4418
MSR	O-HAZE	19.2417	0.0158	7.2059	0.9165	1.7766	2.9659	28.4329
	I-HAZE	15.8653	0.0101	7.4532	0.8639	0.8282	3.1214	31.6590
	Average	17.5535	0.0130	7.3296	0.8902	1.3024	3.0437	30.0456
Three-area	O-HAZE	12.9716	0.0216	6.8442	0.7316	3.6995	3.2322	28.3549
	I-HAZE	10.9020	0.0147	6.8385	0.6331	1.8722	3.2709	35.9219
	Average	11.9368	0.0182	6.8414	0.6824	2.7859	3.2516	32.1384
This paper	O-HAZE	14.4703	0.0222	6.9616	0.7698	3.4288	3.3335	29.6949
	I-HAZE	12.1390	0.0158	7.0985	0.6777	2.1187	3.3006	35.1895
	Average	13.3047	0.0190	7.0301	0.7238	2.7738	3.3171	32.4422

TABLE 4: Execution time of several defogging algorithms.

Defogging algorithms	He	MSR	Three-area	This paper
Execution time	2.1838	0.0691	4.6835	1.8922 + 3.8147 + 8.6052

to those of the comparison algorithm. Combining the subjective visual effect and objective evaluation index, the superiority of the proposed algorithm is illustrated.

In addition, we also make statistics on the average execution time of the algorithm, as shown in Table 4. The algorithm in this paper is mainly divided into three parts. The time of solving the adaptive transmittance and atmospheric light estimation based on the depth of field and using the atmospheric scattering model for defogging is 1.8922 seconds, and the time of calculating the depth of field and bright and dark channel is 3.8147 seconds and 8.6052 seconds, respectively. The algorithm in this paper has achieved good defogging effect, but it takes a long time. Most of the time is spent on the calculation of depth of field and bright and dark channel. The next step will study how to improve the running speed of the algorithm.

5. Conclusions

Aiming at the inapplicable problem of the dark channel prior defogging in the bright area of the image and the blocking effect at the sudden depth of field, this paper proposes an adaptive bright and dark channel combined with defogging algorithm based on the depth of field. On the one hand, the dark channel prior and the bright channel prior are combined to achieve the complementation of the bright and dark areas. On the other hand, based on the depth of field and superpixel segmentation, it realizes the division of content-based abnormity area and the determination of adaptive defogging control factor based on depth of field. It is confirmed that the problem of large transmittance deviation and defogging distortion under content changes and sudden changes in depth of field is well solved. From the experimental results, the algorithm in this paper

has achieved a good defogging effect, with high image clarity, good color recovery, and no block effect at the depth mutation. Under objective indicators, most of the indicators are in the first place, and the average defogging quality is better than common algorithms. Although the algorithm in this paper has achieved good results, there are still some problems to be improved. On the one hand, the shortcomings of the dark channel algorithm are not completely eliminated, and the defogging results are still dark in some scenes. On the other hand, the running speed of the algorithm needs to be improved.

Data Availability

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

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] Y. Xu, J. Wen, L. Fei, and Z. Zhang, "Review of video and image defogging algorithms and related studies on image restoration and enhancement," *IEEE Access*, vol. 4, pp. 165–188, 2016.

[2] W. Wang, X. Wu, X. Yuan, and Z. Gao, "An experiment-based review of low-light image enhancement methods," *IEEE Access*, vol. 8, pp. 87884–87917, 2020.

[3] L. T. Thanh, D. N. H. Thanh, N. M. Hue, and V. B. S. Prasath, "Single image dehazing based on adaptive histogram equalization and linearization of gamma correction," in *25th Asia-Pacific Conference on Communication*, pp. 36–40, Ho Chi Minh City, Vietnam, Nov 2019.

[4] C. K. Sirajuddeen, S. Kansal, and R. K. Tripathi, "Adaptive histogram equalization based on modified probability density function and expected value of image intensity," *Signal Image and Video Processing*, vol. 14, no. 1, pp. 9–17, 2020.

[5] D. Nandi, M. Sarkar, P. R. Sarkar, and U. Mondal, "Empirical wavelet transform based fog removal via dark channel prior," *IET Image Processing*, vol. 14, no. 6, pp. 1170–1179, 2020.

[6] C. Wei, Y. Xu, and Y. Li, "Iterative fusion defogging algorithm based on wavelet transform," *Laser & Optoelectronics Progress*, vol. 12, no. 14, pp. 1–16, 2020.

[7] X. Cai, J. Ma, C. Wu, and H. Xu, "Color image enhancement algorithm based on fuzzy homomorphic filtering," *Computer Simulation*, vol. 37, no. 6, pp. 342–346, 2020.

[8] Q. S. Liu, J. Bai, and F. H. Yu, "An adaptive weight value-based multi-scale Retinex algorithm for color image enhancement," *5th International Conference on Computer Sciences and Automation Engineering*, vol. 42, pp. 609–612, 2015.

[9] J. Chen, Z. Gao, C. Huang, and L. Yang, "Underwater image enhancement algorithm based on Retinex and wavelet fusion," *IOP Conference Series Earth and Environmental Science*, vol. 615, article 01212, 2020.

[10] W. Zhang, L. Dong, X. Pan, J. Zhou, L. Qin, and W. Xu, "Single image defogging based on multi-channel convolutional MSRCR," *IEEE Access*, vol. 7, pp. 72492–72504, 2019.

[11] L. T. Thanh, D. N. H. Thanh, N. M. Hue, and V. B. S. Prasath, "Single image dehazing based on adaptive histogram equalization and linearization of gamma correction," in *25th Asia-Pacific conference on communications*, Ho Chi Minh City, Vietnam, 2019.

[12] L. T. Thanh, D. N. H. Thanh, N. N. Hien, U. Erkan, and V. B. S. Prasath, "Single image dehazing with optimal color channels and nonlinear transformation, Phu Quoc Island, Vietnam, Jan 2021.

[13] D. Wu, X. Zhou, and M. Chen, "Image denoising algorithm based on nonlinear fourth-order PDE," *Journal of Electronic Measurement and Instrumentation*, vol. 31, no. 6, pp. 839–843, 2017.

[14] L. Guo and X. Meng, "Image denoising algorithm based on partial differential equation and mualti-scale analysis," *Journal of Jilin UniversityScience Edition*, vol. 57, no. 4, pp. 882–888, 2019.

[15] K. He, J. Sun, and X. Tang, "Single image haze removal using dark channel prior," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 33, no. 12, pp. 2341–2353, 2011.

[16] Y. Yan, W. Ren, Y. Guo, R. Wang, and X. Cao, "Image deblurring via extreme channels prior," in *IEEE conference on computer vision and pattern recognition*, Honolulu, HI, USA, 2017.

[17] T. Gao, M. Liu, T. Chen, S. Wang, and S. Jiang, "A far and near scene fusion defogging algorithm based on the prior of dark-light channel," *Journal of Xi'an Jiaotong University*, vol. 4, no. 10, pp. 1–9, 2021.

[18] Y. Yan, L. Liu, D. Zhang, and Z. Yang, "Fast single image dehazing combined with adaptive haze estimation," *Optics and Precision Engineering*, vol. 27, no. 10, pp. 2263–2271, 2019.

[19] Y. Yan and X. Lu, "An image dehazing method combining adaptive brightness transformation inequality to estimate transmittance," *Journal of Xi'an Jiaotong University*, vol. 55, no. 6, pp. 69–76, 2021.

[20] Q. Zhu, J. Mai, and L. Shao, "A fast single image haze removal algorithm using color attenuation prior," *IEEE Transactions on Image Processing*, vol. 24, no. 11, pp. 3522–3533, 2015.

[21] D. Fan, X. Ti, Q. Meng, and G. Wang, "An adaptive defogging algorithm based on color attenuation," *Computer Measurement & Control*, vol. 26, no. 9, pp. 200–204, 2018.

[22] X. Guo, P. Sun, X. Lu, and D. Fan, "Image defogging algorithm based on sparse representation," *Journal of Shandong University of Science and Technology Natural Science*, vol. 2020, no. 1, pp. 1–128, 2020.

[23] D. Fan, X. Lu, X. Liu, W. Chi, and S. Liu, "An iterative defogging algorithm based on pixel-level atmospheric light map," *Modelling Identification and Control*, vol. 35, no. 4, pp. 287–297, 2020.

[24] D. Fan, X. Guo, X. Lu, X. Liu, and B. Sun, "Image defogging algorithm based on sparse representation," *Complexity*, vol. 2020, Article ID 6835367,, 2020.

[25] B. P. Kumar, A. Kumar, and R. Pandey, "Region-based adaptive single image dehazing, detail enhancement and pre- processing using auto-colour transfer method," *Signal Processing: Image Communication*, vol. 100, article 116532, 14 pages, 2022.

[26] B. Li, W. Ren, D. Fu et al., "Benchmarking single-image dehazing and beyond," *IEEE Transactions on Image Processing*, vol. 28, no. 1, pp. 492–505, 2019.

[27] B. Cai, X. Xu, K. Jia, C. Qing, and D. Tao, "DehazeNet: an end-to-end system for single image haze removal," *IEEE Transactions on Image Processing*, vol. 25, no. 11, pp. 5187–5198, 2016.

[28] Z. Luan, Y. Shang, X. Zhou, Z. Shao, G. Guo, and X. Liu, "Fast single image dehazing based on a regression model," *Neurocomputing*, vol. 245, pp. 10–22, 2017.

[29] X. Liu, Y. Ma, Z. Shi, and J. Chen, "GridDehazeNet: attention-based multi-scale network for image dehazing," in *IEEE/CVF International Conference on Computer Vision*, pp. 7313–7322, Seoul, Korea (South, Oct 2019.

[30] B. Xie, F. Guo, and Z. Cai, "An image defogging algorithm based on the fog veil theory," *Computer Engineering & Science*, vol. 34, no. 6, pp. 83–87, 2012.

[31] Y. Chen and C. Lu, "Single image dehazing based on superpixel segmentation combined with dark-bright channels," *Laser & Optoelectronics Progress*, vol. 57, no. 16, pp. 161023–161247, 2020.

[32] X. Cheng, X. Liu, X. Dong, M. Zhao, and C. Yin, "Image segmentation based on improved SLIC and spectral clustering," in *Chinese Automation Congress*, pp. 3058–3062, Shanghai, China, 2020.

[33] O. Kwon, "Single image dehazing based on hidden Markov random field and expectation–maximisation," *Electronics Letters*, vol. 50, no. 20, pp. 1442–1444, 2014.

[34] C. O. Ancuti, C. Ancuti, R. Timofte, and C. De Vleeschouwer, "O-HAZE: a dehazing benchmark with real hazy and haze-free outdoor images," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 867–8678, Salt Lake City, UT, USA, 2018.

[35] C. Ancuti, C. O. Ancuti, R. Timofte, and C. D. Vleeschouwer, "I-HAZE: a dehazing benchmark with real hazy and haze-free indoor images," *Lecture Notes in Computer Science*, vol. 11182, pp. 620–631, 2018.

Research Article

An Optimal Scheduling Method for Data Resources of Production Process Based on Multicommunity Collaborative Search Algorithm

Yanlei Yin [iD],[1] **Lihua Wang** [iD],[1] **Jun Tang**,[2] **Wanda Zhang** [iD],[1] **and Hongwei Niu**[3]

[1]*Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, China 650500*
[2]*Technology Center, China Tobacco Yunnan Industrial Co. Ltd., 650233, China*
[3]*Faculty of Machinery and Vehicles, Beijing Institute of Technology, 100081, China*

Correspondence should be addressed to Lihua Wang; kmwanglihua@163.com

Aiming at the problem of low response speed and unbalanced distribution of data resources of production process (DRPP) for the distributed workshop production environment, an optimization scheduling method of DRPP based on a multicommunity cooperative search algorithm is proposed. A heuristic data resource service scheduling framework including a load manager and dynamic scheduling engine is first built to deal with the uncertainty of data resource service response and the imbalance of resource allocation; a core scheduling optimization mathematical model with the objectives: resource service efficiency, reduced response time, and load balancing, is established. Then, a multicommunity cooperative search algorithm for the scheduling model is presented, and the mapping relationship between the particle position vector and resource allocation is established via binary coding. Thus, the optimization algorithm is mapped to discrete data space, and the multicommunity bidirectional driving evolutionary mechanism is used to realize the cooperative and interactive search between common and model community, which enhances the adaptability of the algorithm to dynamic random scheduling tasks. Finally, the effectiveness of the proposed method is verified by an example of multiprocess quality prediction service scheduling in silk production process, which provides an effective means for solving the complex scheduling problem of production process data.

1. Introduction

The intelligent workshop integrates modern sensing technology, network technology, automation technology, and other advanced technologies, and a large number of intelligent equipment such as sensors and data acquisition devices have been put into use in the workshop [1–3]. And thus, the production workshop has become the collection center of information flow, material flow, and control flow. In the process of product production, a large amount of production, environment, status, and equipment operation data are generated at an unprecedented speed, thus forming workshop big data, which presents the new characteristics of multitask, cross-process, heterogeneous, and polymorphic. However, data has no subjective initiative. To realize the real-time perception and prediction of the production

process, we must first realize the scheduling and optimization of "data," that is, the scheduling and optimization of production process data. It is noted that the traditional manufacturing mode, data of information flow, material flow, and control flow are still isolated from each other in each stage of production execution, and it is difficult to form a joint force due to the lack of effective data resource scheduling mechanism, which restricts the further improvement of production efficiency and system intelligence level [4]. Therefore, the research on on-demand scheduling of workshop data resources is one of the core problems of intelligent manufacturing in production workshop.

Recently, most scholars at home and abroad focus on the methods and algorithms of workshop data collection, analysis and mining, such as machine vision preprocessing algorithm [5], neural network prediction algorithm [6, 7],

intelligent decision algorithm [8], and multiobjective optimization algorithm [9]. However, the data have no subjective initiative; the data-based analysis and processing algorithm can not actively serve the business needs such as perception, decision-making, and execution of the production process; and the current research has not comprehensively considered the coupling and impact among demand, service, resources, and energy efficiency in the production process.

However, the research of domestic and foreign scholars on workshop process production scheduling and collaborative job scheduling has a good reference for the development of this work. Considering workshop process and production scheduling, literature [10] based on the dependence of the production planning, and scheduling problem of continuous production line on timing, a repair and optimization solution is proposed to solve the problem of energy efficiency in the production process. For the scheduling problem of complex products in multiworkshop production, literature [11] studies the characteristics of BOM structure and process route of complex products. Based on the construction of multilevel process network diagram, an improved particle swarm optimization algorithm is used to ensure the effectiveness of the algorithm search path. Literature [12] considers the problems of process connection and blocking of prefabricated parts in the process of workshop assembly line operation and establishes a scheduling model to minimize the total penalty cost of advance and delay, which improves the production efficiency of Prefabrication Yard. Literature [13] uses a machine learning method to assign jobs based on the priority rules of the decision tree as the scheduling method, which shows good performance in the case scenario with completion goal and total delay goal. Considering from workshop production collaborative job scheduling, Literature [14] estimates the process processing time in the production process through machine learning and uses the estimated processing time to schedule and optimize parallel machines, which reduces the maximum completion time by about 30% on average. Aiming at the optimal comprehensive production and transportation plan of a group of parallel batch machines, literature [15] constructs a 0-1 mixed integer programming model, solves the model, and completes the comprehensive scheduling through an improved genetic algorithm, which reduces the transportation cost. The above research provides an idea for this paper to realize on-demand scheduling data resources for production process.

On the other hand, it should be noted that the workshop production process involves multiprocess cross-production line business collaboration and business requirements. In the process of data resource scheduling, we should not only consider the association and cooperation relationship between different production tasks but also consider the transmission time of data resources between different production tasks, in particular, the uncertainty of concurrent service access affecting demand response, and the impact of these dynamic and uncertain factors on the balance of data resource allocation [16–18].

This paper is concerned with production-oriented data resources scheduling, thus transforming DRPP into a service, and finally into economic benefits. Consequently, this study is to integrate the load manager and dynamic task scheduling engine, and combine them with scheduling processes to form a scheduling scheme, so as to provide intelligent support for production process. Following this idea, this paper is organized as follows: the heuristic data resource service scheduling framework is constructed in Section 1. The problem to be studied and the scheduling mathematical model are proposed in Section 2. Section 3 is devoted to establish the asynchronous parallel scheduling strategy and optimization method, simulation results are given in Section 4, and conclusions are made in Section 5.

2. DRPP Scheduling Process Analysis

According to the execution status of the DRPP in the scheduling center, the data in the resource pool are mobilized to form the optimal execution scheme of tasks. DRPP scheduling is one of the key links in the production decision process. Figure 1 presents a framework of DRPP scheduling that includes decomposition of business requirements, service task analysis, dynamic scheduling of DRPP, load monitoring, and service task execution. Firstly, during production operation, different processes send requests to the scheduling center according to task execution requirements. By queuing, merging, and analyzing the service requirements, the task analyzer degrades the vague and miscellaneous service tasks to form a set of low-granularity service tasks that can be directly served by DRPP. Secondly, the dynamic scheduler preliminarily matches the DRPP according to the task request and then matches the execution task characteristics with the static and real-time attributes of the DRPP to obtain the DRPP set that meets the current production business requirements. The load supervisor of the scheduling center dynamically adjusts the DRPP by monitoring the operation and load of the DRPP in the business process and solves the service interruption caused by uncertain events to ensure the accomplishment of the service process. Finally, the DRPP scheduling engine uses the integrated intelligent optimization algorithm to combine and match the state information of the DRPP and the real-time information of the service task to form the optimal resource service allocation scheme and submits it to the center for execution, so as to complete the scheduling process of the resource service.

Throughout the entire scheduling process, multiservice tasks are executed interactively, and there is a complex relationship between tasks. With the dynamic growth of production business scale, the response time of service tasks must be considered. Additionally, a large number of dynamic and uncertain factors will seriously affect the ability and effectiveness of DRPP service scheduling. The traditional resource scheduling method has low search efficiency and accuracy, which can easily lead to the problems of low response speed and the uneven distribution of resources in the service process. It is difficult to adapt to the allocation of STRs on demand. Therefore, this paper considers
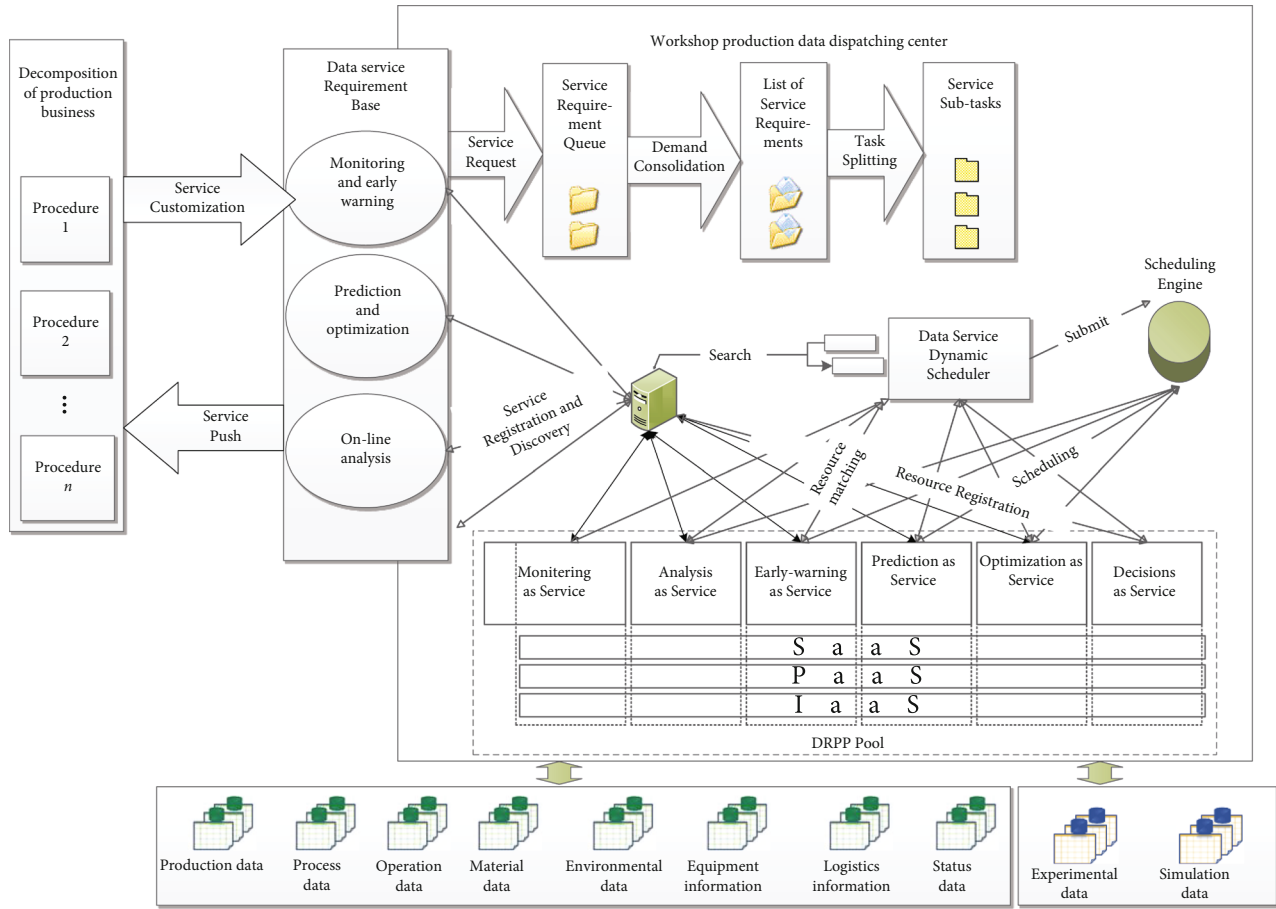
FIGURE 1: Scheduling process of DRPP.

improving service efficiency while solving the problem of the unreasonable allocation of DRPP caused by the uncertainty of service response and the unbalanced load of nodes.

3. The Multiobjective Optimal Scheduling Model of DRPP

3.1. The Response Uncertainty Modeling. The invocation relationship of DRPP in the business process is complex, especially when a data resource service is invoked by multiple business processes, and these business processes run simultaneously; there will be concurrent access. In different service scenarios, there are some fluctuations and uncertainties in the access frequency, concurrency probability, and response time of users in the entity industry. There are certain fluctuations and uncertainties in the access frequency, concurrency probability, and response time of different business links in the production process, such as sequence, selection, parallelism, and the cycle of business processes [19, 20]. Therefore, the uncertainty of the service response of DRPP can be described by service access frequency.

In the following formula, k represents the number of service tasks. l represents the total DRPP resources. p represents the process of any service tasks. θ represents the probability of performing service task i. μ represents the probability of

invoking DRPP j for the service task i. ε represents the probability of a service business being accessed. γ represents the probability of any process branch being selected.

Assuming that there are j DRPP for I subtasks to call when the scheduling center performs a certain service task, and these subtasks are completed on a specific node according to the service process; then, the probability that the process of the service task is executed is

$$\varepsilon = \sum_{j=1}^{l} \sum_{i=1}^{k} \theta\mu \, 1 \le i \le I \, 1 \le j \le J. \tag{1}$$

When multiservice tasks are executed interactively, concurrent access often occurs in service invocation. Accomplishing a task involves invoking multiple resource service processes. When there is a selection structure in the process and a process branch covers concurrent access services, the probability of concurrent access to services is as follows:

$$P_{ca} = \sum_{p=1}^{p} \varepsilon\gamma \, 1 \le p \le L. \tag{2}$$

When all the selected branches in the service process cover the current concurrent access service, the probability

of concurrent access to the services is

$$P_{ca} = \sum_{p=1}^{p} \varepsilon \, 1 \leq p \leq L. \tag{3}$$

3.2. The Modeling of Unbalanced Resource Allocation. Suppose a service task $\Omega = \{t_1, t_2, \cdots, t_n\}$ needs to call k subtasks to complete, the total number of the DRPP that can provide services is l. These subtasks are completed in different tasks node according to the service process and resource requirements. The expected completion time of task t_i invoking technology resource r_j is defined as

$$T_{ij} = \frac{IL_i}{ES_j}, \tag{4}$$

where IL_i is the total instruction length of service task t_i and ES_j is the execution speed that the DRPP r_j are distributed invocation.

The average load of k different service tasks scheduling l DRPP is defined as the quotient between the total instruction length of k service tasks and the total execution speed of l data resources distributed scheduling, i.e., the completion time of the total service task is

$$Tt = \frac{\sum_{i=1}^{k} IL_i}{\sum_{j=1}^{l} ES_j}. \tag{5}$$

For the above scheduling scheme, the load balancing of service resources invoked can be defined as

$$\sigma = \sqrt{l^{-1} \sum_{j=1}^{l} \left(Tt_j - Tt\right)^2}, \tag{6}$$

where Tt_j represents the completion time of the service task j. Obviously, a smaller σ indicates a more balanced task load service scheduling.

3.3. The Multiobjective Optimal Scheduling Model. Considering the DRPP response and the unbalance of resource allocation, this paper established a multiobjective optimal scheduling mathematical model including the service efficiency, response time, and load balance of resource invocation.

In the production process, if e_j^i represents the state of service node i performing subtask j, N represents the subtasks, and M represents the total number of DRPP provided by service tasks; then, the state set of resource services is $E = (e_j^i)_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$. The set of service efficiency is $Se = \{se_j^i\}_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$, the response frequency set is $Rt = \{rt_j^i\}_{M \times N}$, $1 \leq i \leq M$, $1 \leq j \leq N$, and the load balancing set is $\delta = \{\sigma_j^i\}_{M \times N}$ $1 \leq i \leq M$, $1 \leq j \leq N$. In these equations, se_j^i is the service efficiency of service node i executing scheduling task j, rt_j^i is the response time of executing

resource scheduling task j for service node i, and σ_j^i is the service efficiency load balancing of resource scheduling task j for service node i. The service status of any resource service node executing a task can then be expressed as

$$e_j^i = \left\{ se_j^i, rt_j^i, \sigma_j^i \right\}. \tag{7}$$

The set of DRPP mapped by N service tasks is $X = \{x_1, x_2, \cdots, x_N\}$, where x is a DRPP invoked for a service subtask. Thus, the mathematical model of multiobjective optimal scheduling considering the uncertainty of service response and the imbalance of resource allocation is as follows:

$$\begin{cases} \min F(x) = (X, E) = (Se(x), Rt(x), \delta(x)), \\ \quad x_j \in R \text{ and } x \leq M, \\ \quad se_j^i \geq se_{\min}, \\ s.t. \\ \quad 0 \leq rt_j^i \leq rt_{\max}, \\ \quad 0 \leq \sigma_j^i \leq \sigma_{\max}, \end{cases} \tag{8}$$

where $se_{\min}$ is the minimum service efficiency value in line with business requirements, $rt_{\min}$ is the maximum service response time that a DRPP node can take, and $\sigma_{\max}$ is the highest load balancing of a DRPP node.

Set the weights of the service efficiency, load balancing, and response frequency of user requesting DRPP as ω_{Se}, ω_{Rt}, and ω_δ, respectively, and $\omega_{Se} + \omega_{Rt} + \omega_\delta = 1$.

The optimal scheduling model of DRPP considering the target weight of user demand is as follows:

$$F(x) = \omega_{Se} Se(X) + \omega_{Rt} Rt(X) + \omega_\delta \delta(X). \tag{9}$$

4. The Optimal Scheduling Algorithms of DRPP Based on Multicommunity Collaborative Search

4.1. The Evolution Model of Multicommunity Cooperative Network. The basic particle swarm optimization (PSO) algorithm is a single-community optimization model with global optimal particles as its core, which cannot solve the mixed and changeable scheduling problem very well. If this model is extended to task-related multicommunity cooperative optimization, the evolutionary information interaction and association will be generated among these communities, and then, a multicommunity cooperation network (MCCN) with high adaptability to the task will be formed [21, 22]. From a mathematical point of view, a network can be regarded as a combination of a vertex set and edge set. To better describe MCCN and establish its evolution model, the following definitions are first provided.

Definition 1. The threshold FT for community type determination is

$$\text{FT} = \frac{\sum_{i=1}^{n} F_i}{n}, \tag{10}$$

where F_i is the global optimal fitness of community i and n is the number of communities in the cooperative network.

According to the threshold FT of community type determination, the particle community in the collaboration network can be divided into the model community and common community. If F_i satisfies the criterion $F_i \geq \text{FT}$, the community has a strong ability of local optimization, which can be divided into model communities and recorded as MC_i. On the contrary, if F_i satisfies the criterion $F_i < \text{FT}$, the community has a strong ability of global exploration, so it can be divided into common communities and recorded as CC_i.

Definition 2. Let the cooperative search activity among different communities be a binary group (C, R), where $C = \{c_1, c_2, \cdots c_j, \cdots c_n\}$ $(1 \leq j \leq n)$ is the sequence of communities participating in the cooperative search activity and $R : C \times C$ is the interdependency among communities in the search process. $\forall (r_s : \langle c_i, c_j \rangle) \in R, (s = 1, 2, 3)$ is called the cooperative relationship unit, where r_1 represents the cooperative relationship between a model community and a common community, r_2 is the cooperative relationship between any two model communities, and r_3 is the cooperative relationship between any two common communities. The number of cooperative units among different communities in a cooperative relationship set is called the module of the cooperative relationship set, which is recorded as $\|R\|$.

Generally, if $r_s(c_1, c_2) = 1$, there is an edge between two cooperative communities, and the more cooperative relationship units, the greater the edge weight between two different nodes. If there is no cooperative relationship between different communities, then $r_s(c_1, c_2) = 0$.

Definition 3. Let $\omega_{i,j} = \sum_{s=1}^{\|R\|} r_s(c_1, c_2)$ be the cooperative weights among different communities in MCCN, where cooperative weight between c_i and c_j is also called the edge weight of MCCN.

To complete the comprehensive quantitative evaluation of community nodes, the evaluation indexes of the optimum value $g_{\text{best}i}$ of community nodes are introduced: collaboration distance H_i and responsivity e_i.

Definition 4. Collaboration distance. The global optimal value $G_{\text{best}i}$ of the community i is, respectively, compared with the individual optimal position $p_{\text{best}j}$ of the m particles, and the absolute value is obtained, that is to say, the cooperative distance of the global optimal value $G_{\text{bset}i}$ is $H_i = (h_1, h_2, \cdots, h_m)$.

Definition 5. Responsivity. The threshold D of the qualified distance is set. According to the formula $ev_i = \begin{cases} 0, h_i > D, \\ 1, h_i < D, \end{cases}$, the response value of the community particle to the optimal value $g_{\text{best}i}$ of the node can be obtained by traversing the cooperative distance H_i, and then, the responsivity e_i of the global optimal value $G_{\text{best}i}$ can be obtained by adding the response values in sequence.

Definition 6. Community node strength. In MCCN, the strength of the community nodes is defined as s_i

$$s_i = \sum_{c_j \in U_j} \omega_{ij} + e_i \tag{11}$$

where ω_{ij} is the cooperative weight between the community node c_i and c_j, e_i is the responsivity of the community node, and U_i is the neighborhood of the community node c_i, and it satisfies

$$U_i \left\{ c_j \left| \bigvee_{s=1}^{\|R\|} r_s(c_i, c_j) \neq 0 \right. \right\}. \tag{12}$$

Generally, MCCN can be represented by its adjacency matrix as $A(G)_{n \times n} = (B)_{n \times n}$. If $E_{n \times 1} = [e_1, e_2, \cdots, e_n]$ is the responsivity matrix of MCCN, then the node strength matrix is as follows: $S_{n \times 1} = [\omega_{11} A(G)_{11} + \omega_{12} A(G)_{12} + \cdots \omega_{1n} A(G)_{1n} + e_1, \omega_{21} A(G)_{21} +$

$\omega_{22} A(G)_{22} + \cdots \omega_{2n} A(G)_{2n} + e_2, \cdots, e_1, \omega_{n1} A(G)_{n1} + \omega_{n2} A(G)_{n2} + \cdots \omega_{11} A(G)_{11} + e_n]$.

Definition 6 shows that the strength of community nodes not only takes into account the cooperative weights among the nodes of the community but also the optimization of the particles within the node itself. It is a comprehensive evaluation of the community's local information and the ability of the community itself, which can better reflect the community's ability to seek optimal guidance in the entire cooperative network.

Therefore, MCCN can be represented by undirected weighted graphs $G(C, R, W, S)$. $C = \{c_1, c_2, \cdots, c_n\}$ represents different types of cooperative community node set, $R = \{r_1(c_1, c_2), r_2(c_1, c_2), \cdots, r_k(c_1, c_2), \cdots, r_s(c_1, c_2)\}$ represents cooperative relationship edge set, $W = \{\omega_{11}, \omega_{12}, \cdots \omega_{ij}, \cdots \omega_{nn}\}$ $(1 \leq i, j \leq n)$ is the cooperative edge weight set among, and $S = \{s_1, s_2, \cdots, s_n\}$ is the strength set of community nodes, where s_i is the value of the i-th row of the node strength matrix, representing the attributes of community nodes to measure their search ability. By Definition 6, MCCN can be expressed by adjacent augmentation matrix M as follows: $M(G) = (B, E)_{n \times (n+1)}$. The evolution model of the MCCN cooperative network is therefore

$$B_{n \times n} = B[i,j]_{n \times n} = \begin{cases} \omega, & \bigvee_{s=1}^{\|R\|} r_s(c_i, c_j) = 1, r_s \in R, \\[3mm] 0, & \bigvee_{s=1}^{\|R\|} r_s(c_i, c_j) = 0, \text{ or } i = j, r_s \in R. \end{cases}$$

(13)

On this basis, the asynchronous parallel search strategy among different communities is formulated to reduce the communication between communities, and the efficient search is realized through the driving evolution mechanism to improve the optimization ability of the algorithm to the task scheduling. The rules of multi-population coevolution are as follows.

Rule 1. Evolutionary rules within microbial communities. In the process of multicommunity coevolution, the particles in a single community can be iteratively optimized according to formula (13) for speed and location updating, and the global optimum value can be generated within the community.

$$\begin{cases} v_{id}^{t+1} = \omega \cdot v_{id}^t + c_1 \cdot r_1 \cdot (P_{id}^t - x_{id}^t) + c_2 \cdot r_2 \cdot (P_{gd}^t - x_{id}^t), \\[2mm] x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, i = 1, 2, \cdots, m, d = 1, 2, \cdots, D, \end{cases}$$

(14)

where t is the number of iterations of particle search, ω is the inertial weight, $c_1 = c_2 = 2$ is the acceleration constant, and r_1 and r_2 are two random functions varying in the range of $[0, 1]$.

Rule 2. Driving coevolution rules between communities.

Rule 2.1. $\forall (r_1 : \langle CC_i, MC_j \rangle) \in R, \exists g_{best i} = \max \{g_{best 1}, g_{best 2}, \cdots, g_{best m}\}, G_{best j} = \min \{G_{best 1}, G_{best 2}, \cdots, G_{best n}\}$, and $g_{best i} \geq G_{best j}$. The common community is CC, and the model community is MC. The particulate CC_i in CC enters MC, and the last community MC_j in MC is eliminated. After introducing the model learning factor p_n into the internal evolution rules of CC, the new iterative evolution formula is as follows:

$$\begin{cases} v_{id}^{t+1} = \omega \cdot v_{id}^t + c_1 \cdot r_1 \cdot (P_{id}^t - x_{id}^t) + c_2 \cdot r_2 (P_{gd}^t - x_{id}^t) + c_3 (P_{nd} - x_{id}^t), \\[2mm] x_{id}^{t+1} = x_{id}^t + v_{id}^{t+1}, i = 1, 2, \cdots, m, d = 1, 2, \cdots, D, \end{cases}$$

(15)

where $P_{nd} = \sum_{i=1}^n G_{best i}/n$ and c_3 is a random function and satisfies the convergence constraints $c_1 r_1 + c_2 r_2 + c_3 \in [0, 4]$.

Rule 2.2. $\forall (r_2 : \langle MC_i, MC_j \rangle) \in R, \exists$ the community node strength S_{MCi} satisfies $S_{MCi} \geq S_{MCj}$ for any S_{MCj}.

$\Rightarrow$ Global optimum value of model community: $PG = G_{best i}$.

Rule 2.3. $\forall (r_3 : \langle CC_i, CC_j \rangle) \in R, \exists$ the community node strength S_{CCi} satisfies $S_{CCi} \geq S_{CCj}$ for any S_{CCj}.

$\Rightarrow$ Global optimum value of common community: $Pg = g_{best i}$.

4.2. The Coding Strategy for Optimal Scheduling of DRPP.

Particle swarm optimization (PSO) is a computational model for real continuous space, and it is difficult to solve the task scheduling problem in discrete space [23]. Therefore, the binary system is used to encode the speed and position of particles, and the mapping from the particle swarm optimization algorithm to discrete space, and from the particle search space to the optimal scheduling scheme, is realized by reconstructing the particle expression.

In the above algorithm, an n row, n column matrix $X : n \times n$ is defined as the position vector matrix of particles. The rows represent the situation of providing STR when any service task is executed, the columns indicate the distribution of service tasks in the scheduling process, and any particle represents the potential solution of the scheduling problem. The coding of the particle position is as follows:

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1n} \\ x_{21} & x_{22} & \cdots & x_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nn} \end{bmatrix},$$

(16)

where $x_{ij} \in \{0, 1\}, \sum_{j=1}^n x_{ij} = 1$.

According to the coding scheme, each row of the location matrix X has one and only one element value is 1, which indicates that DRPP are allocated to service task R. Each DRPP can be invoked by multiple service tasks simultaneously, and the execution of any scheduling task cannot be interrupted.

The defined speed $V : n \times n$ is shown in equation (16), which represents the basic exchange order of particle's assignments to the execution of tasks.

$$V = \begin{bmatrix} v_{11} & v_{12} & \cdots & v_{1n} \\ v_{21} & v_{22} & \cdots & v_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ v_{n1} & v_{n2} & \cdots & v_{nn} \end{bmatrix},$$

(17)

$v_{ij} \in \{0, 1\}, v_{ij} + v_{ji} = 0 \text{ or } 1.$

The exchange operations of addition, subtraction, multiplication, and division in the algorithm are defined as Θ, θ, $\otimes$, and $\oplus$, respectively. The specific operation rules are as follows:

(1) $A \cdot \theta \cdot B$: represents $\exists x_{ij} = v_{ij} \Rightarrow x_{ij} = v_{ij} = 0$ in position matrix A and velocity matrix B; on the contrary, it is 1; $\exists x_{ij} = v_{ij+n} = 1 \Rightarrow v_{ij+n} = 0$

(2) $A \Theta B$: indicates $\exists x_{ij} = v_{ij} \Rightarrow x_{ij} = v_{ij} = 0$ in position matrix A and velocity matrix B; the other elements are randomly chosen as 0 or 1

(3) $c_i \otimes B$: indicates whether the particle performs a Θ operation or not with matrix B according to the

corresponding probability value of the random number c_i

(4) $A \oplus B$: represents $\forall x_{ia} = 1$, $x_{jb} = 1$, $\exists v_{ij} = 1 \Rightarrow x_{ib} = 1$, $x_{ja} = 1$ in position matrix A and velocity matrix B. According to the above definition of switching operation rules, formula (10) can be updated as follows:

$$\begin{cases} v_{id}^{t+1} = v_{id}^t \Theta c_1 \otimes \left(P_{id}^t \cdot \theta \cdot x_{id}^t \right) \Theta c_2 \otimes \left(P_{gd}^t \cdot \theta \cdot x_{id}^t \right), \\ x_{id}^{t+1} = x_{id}^t \oplus v_{id}^{t+1}, i = 1, 2, \cdots, m ; d = 1, 2, \cdots, D \end{cases}$$

$$(18)$$

The coding scheme is simple and feasible and thus meets the requirements of multiservice task scheduling of DRPP. It also clearly describes the mapping relationship between the particle population evolution space and the service task scheduling scheme, thus avoiding repeated searches in the process of particle evolution.

4.3. The Optimal Scheduling Algorithms of DRPP Based on Multicommunity Collaborative Search. Based on the multicommunity cooperative search algorithm and its coding scheme, the optimal scheduling process of multiservice tasks for distributed DRPP is shown in Figure 2. The specific steps are as follows.

Step 1. Initialization of population particles. According to the encoding strategy between the particle search space and the task scheduling scheme described in Section 3.2, the initialization of n communities is carried out, and a random location (DRPP allocation scheme) and speed of population particles are given. The number of communities, the number of iterations of particles within the community members, and the acceleration coefficient of particles and the inertia weight coefficient are set.

Step 2. Initialized population particles are evenly distributed into the q process to form a community of size int (n/q). Residual particles are randomly allocated to the q process. The fitness of each particle in the q community is calculated according to the comprehensive optimization scheduling function constructed in Section 3.3.

Step 3. Asynchronous parallel evolutionary computation is performed by running each community separately in the q process.

Step 4. Calculate the fitness values F_i of each community and divide all communities into either the model community or common community according to the threshold value.

Step 5. According to the interactive evolution mechanism between different particle populations in Section 3.1, the position and velocity of particles are updated according to formula (14), and the global optimal locations of the model and common communities are saved to the optimal storage area.

Step 6. If all the particle populations satisfy the search termination condition, then the algorithm ends, the global optimal solution is obtained from the storage area, and the
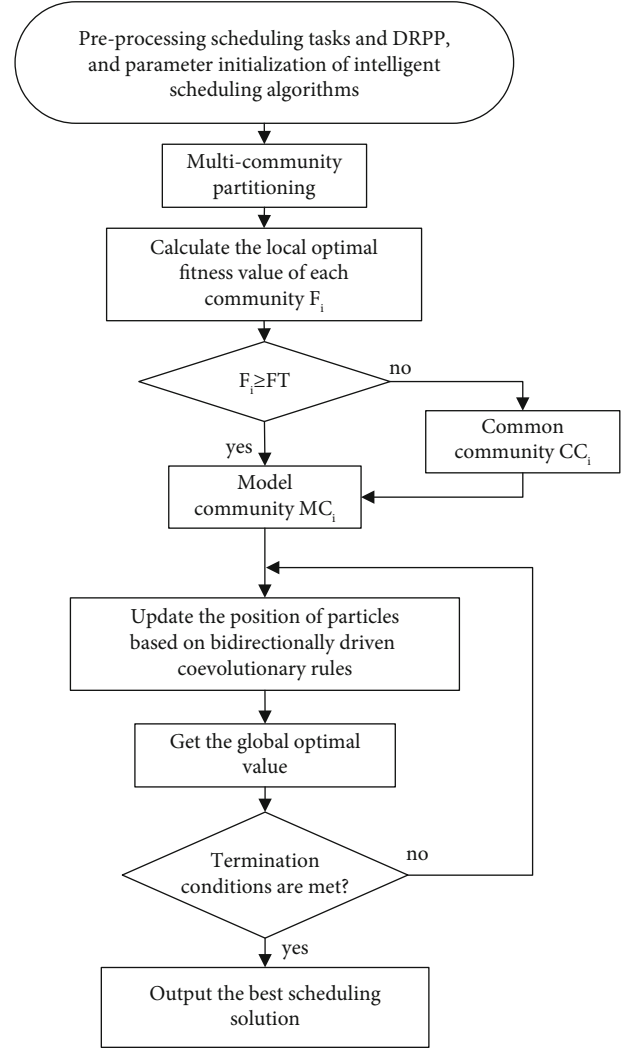
Figure 2: Flow chart of scheduling optimization.

optimal scheduling scheme is output; otherwise, it will return to step 5.

5. Application Cases and Analysis

In this section, a DRPP scheduling case for the silk production line quality prediction and early warning service are given to validate the proposed model and algorithm. As shown in Figure 3, the quality prediction and early warning service include the " single operation quality prediction and early warning," " multiprocess quality prediction and early warning," " quality prediction and optimization of the whole production line," and many other services, where each service activity needs to invoke DRPP using service tasks, such as the modle, standards, algorithm, and component to provide on-demand service for different business links of the the tobacco production line.

For the task of quality prediction and optimization of silk drying process service in the service platform, it is divided into five subtasks: online data reading, prediction algorithm call, online real-time prediction, prediction result

FIGURE 3: The DRPP service platform.

TABLE 1: Particle encoding diagram.

Sub-tasks	Resources					
	X_1	X_2	X_3	X_4	X_5	X_6
t_1	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}
t_2	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}
t_3	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}
t_4	x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	x_{46}
t_5	x_{51}	x_{52}	x_{53}	x_{54}	x_{55}	x_{56}
t_6	x_{61}	x_{62}	x_{63}	x_{64}	x_{65}	x_{66}

analysis, optimization algorithm call, and optimization parameter return, which is recorded as task set $\Omega = \{t_1, t_2, \cdots, t_5\}$. And then, the available DRPP set corresponding to task set Ω is $X = \{x_1, x_2, \cdots, x_N\}$. Following the idea proposed in Sections 1 and 2, the DRPP invoking process can be implemented as follows:

(1) Coding settings

Define the location vector of the particle in the scheduling algorithm as the matrix $X : 5 \times 5$, as shown in Table 1. Row i represents the allocation of service task t_i, and column j represents the service situation of DRPP; if $x_{ij} = 1$, it means that the task t_i is served by data resources X_j. At this time, each particle represents a service task scheduling scheme.

(2) Scheduling algorithms

To verify the effectiveness of the presented scheduling method, the simulation experiment is carried out on the service platform based on the Xeon E5-2609V2 processor and RAM64G, and the service efficiency, response time, and load balancing in the process of multiservice task scheduling are collected as sample data. For comparison, the previously reported ERTPSO algorithm [24], DPSO algorithm [25], LAPSO algorithm [26], and the M-CBDCSM algorithm proposed in the present study are used to solve the optimal

(a) $n = 0$

(b) $n = 1$

(c) $n = 2$

(d) $n = 3$
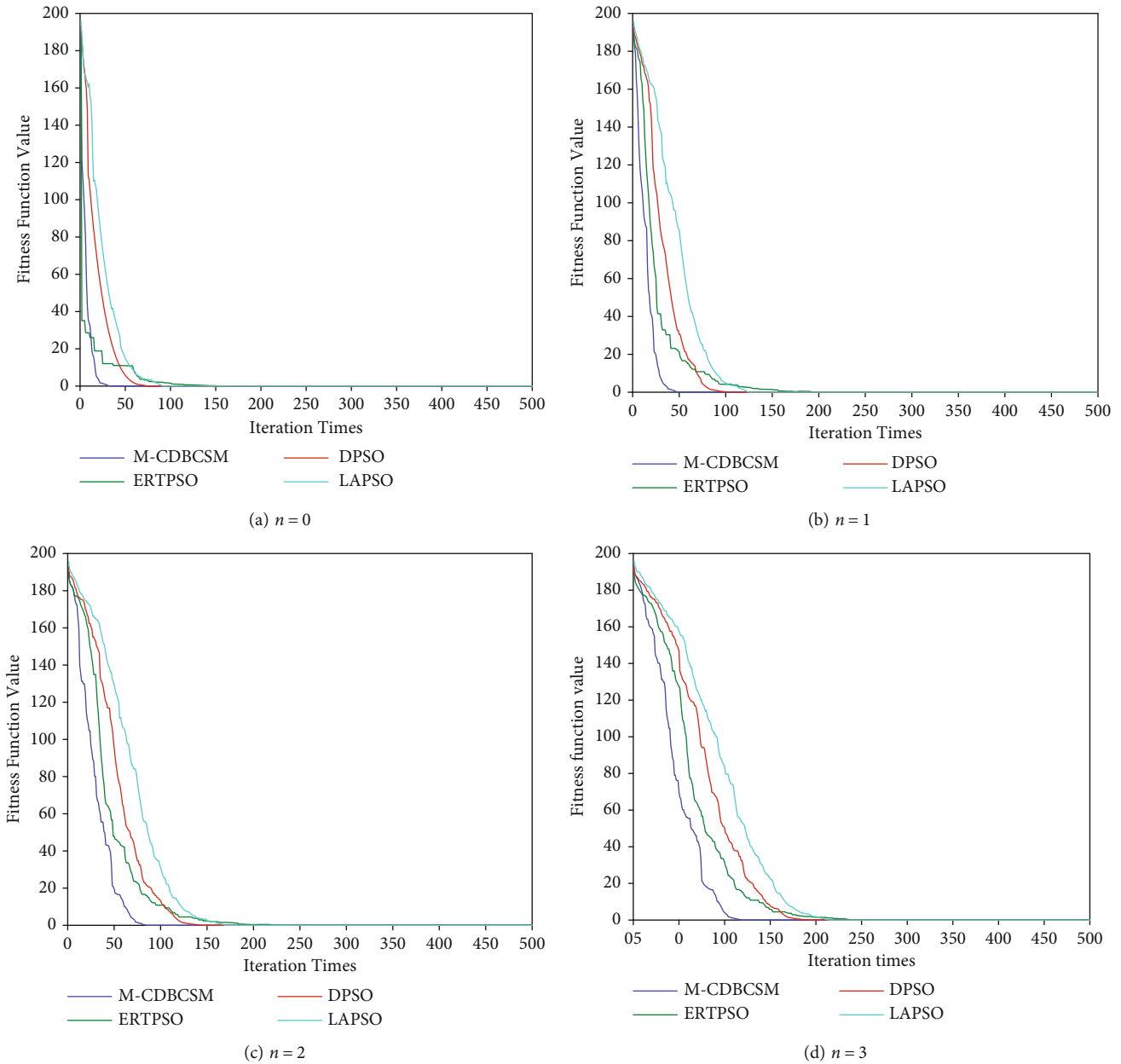
FIGURE 4: Comparison of simulation experiments.

TABLE 2: Comparative experimental results of different task sizes.

Task scale	Convergent algebra	Optimal value	Standard deviation	Average error
5	15	60	$1.73e - 011$	$1.21e - 012$
10	15	130	$2.81e - 011$	$1.61e - 012$
15	17	185	$1.58e - 010$	$2.57e - 012$
20	18	255	$3.95e - 010$	$3.43e - 012$
25	18	300	$6.12e - 010$	$2.69e - 011$

($n = 0$, iteration $= 200$).

scheduling scheme for the engine fault identification and maintenance service. The simulation parameters are specified as the population size $q = 30$-500, the dimension 30, the evolutionary algebra $i = 500$, the inertia weight $\omega = 1.1$, and the acceleration constant $c_1 = c_2 = 2$. Moreover, a random function rand () is introduced into the service response time Rt_d; $Rt_d = Rt + n \cdot$ rand (), where rand () varies randomly in a range $[0, 1]$, and $n = 0, 1, 2, 3$. All the simulation experiments were carried out 500 times, and other parameters are the same to those used in the relevant literature.

Simulation results of the relevant algorithm are provided in Figure 4. It is shown that the multicommunity cooperative search algorithm can better adapt to the random changes of service response time in the process of

multiservice task scheduling, in particular for the interactive evolutionary rules between communities are selected adaptively, the algorithm converges to the global optimal value quickly and stably. In particular, the optimal scheduling scheme can be found before 50 generations under different search conditions corresponding to the discrete particle swarm optimization (DPSO). Especially in the face of dynamic and random multitask scheduling, hybrid genetic algorithm is difficult to adaptively carry out individual mutation and cross-operation and then can not track the dynamic change of service scheduling, which makes it difficult to avoid premature phenomenon under high random search conditions, and its algorithm performance is far lower than multicommunity cooperative search algorithm.

To further validate the strong adaptability of the algorithm in the face of multiple scheduling tasks, the experiments of large population based on multi scheduling tasks is also simulated. For this purpose, the population and the scheduling tasks can be given as $p = 1000$, $t = 5\text{-}20$. The corresponding results are provided in Table 2. Although the number of service tasks is increasing, the convergence rate of multi community cooperative search algorithm does not decline significantly in the process of scheduling. From aforementioned results, one can conclude that the proposed algorithms in this paper can achieve better performance for large population and tasks-varying parameters in terms of convergence speed and steady-state errors.

6. Conclusion

This paper addresses the multitask adaptive scheduling of DRPP. A heuristic scheduling framework are employed to deal with the uncertainty of DRPP service response and the imbalance of resource allocation. The load manager and dynamic scheduling engine are employed to approximate the uncertainty of scheduling service. Moreover, we propose novel cooperative search algorithm of the task scheduling model driven by scheduling objectives with service efficiency, reduced response time, and load balancing, so that fast scheduling convergence can be proved even in the dynamic random sense. The proposed scheduling schemes are robust against dynamic random disturbances. To guarantee the discrete data space of optimization algorithm, a binary coding strategy to map the particle position vector to resource allocation is introduced. Simulation examples are provided to verify the efficacy of the proposed algorithm.

Data Availability

The production process data used to support the findings of this study have not been made available because the enterprise of the data source requires confidentiality.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

[1] W. Xu, S. Guo, X. Li, C. Guo, R. Wu, and Z. Peng, "A dynamic scheduling method for logistics tasks oriented to intelligent manufacturing workshop," *Mathematical Problems in Engineering*, vol. 2019, 18 pages, 2019.

[2] Z. F. Liu, W. Chen, C. B. Yang, Q. Cheng, and Y. S. Zhao, "Intelligent manufacturing workshop dispatching cloud platform based on digital twins," *Computer Integrated Manufacturing Systems*, vol. 25, no. 6, pp. 1444–1453, 2019.

[3] S. Huang, Y. Guo, and S. Zha, "Review on internet-of-manufacturing-things and key technologies for discrete workshop," *Jisuanji Jicheng Zhizao Xitong/Computer Integrated Manufacturing Systems, CIMS*, vol. 25, no. 2, pp. 284–302, 2019.

[4] M. K. Lim, W. Xiong, and Z. Lei, "Theory, supporting technology and application analysis of cloud manufacturing: a systematic and comprehensive literature review," *Industrial Management & Data Systems*, vol. 120, no. 8, pp. 1585–1614, 2020.

[5] X. Sun, G. Gao, B. Tian, B. Li, S. Zhang, and J. Huang, "Intelligent recognition of production date based on machine vision," in *Journal of Physics: Conference Series, Volume 1267, 2019 3rd International Conference on Artificial Intelligence, Automation and Control Technologies (AIACT 2019)*, Xi'an, China, April 2019.

[6] X. Ye, X. Wu, and Y. Guo, "Real-time quality prediction of casting billet based on random forest algorithm," in *2018 IEEE International Conference on Progress in Informatics and Computing (PIC)*, Suzhou, China, 2019.

[7] L. Zhiyong, Z. Rou, Z. Jie, and C. Ting, "Research on optimization of centrifugal process parameters based on support vector machine," in *2020 39th Chinese Control Conference (CCC)*, Shenyang, China, 2020.

[8] H. Li, C. Wang, S. Jiang, S. Liu, Y. Rong, and X. Li, "The study of intelligent scheduling algorithm oriented to complex constraints and multi-process roller grinding workshop," *Advances in Mechanical Engineering*, vol. 12, no. 11, 2020.

[9] X. P. Yang and X. L. Gao, "Optimization of dynamic and multi-objective flexible job-shop scheduling based on parallel hybrid algorithm," *International Journal of Simulation Modelling*, vol. 17, no. 4, pp. 724–733, 2018.

[10] J. Liang, Y. Wang, Z. H. Zhang, and Y. Sun, "Energy efficient production planning and scheduling problem with processing technology selection," *Computers & Industrial Engineering*, vol. 132, pp. 260–270, 2019.

[11] L. Qiao, Z. Zhang, and Z. Huang, "A scheduling algorithm for multi-workshop production based on BOM and process route," *Applied Sciences*, vol. 11, no. 11, p. 5078, 2021.

[12] Y. Dan, G. Liu, and Y. Fu, "Optimized flowshop scheduling for precast production considering process connection and blocking," *Automation in Construction*, vol. 125, article 103575, 2021.

[13] F. Benda, R. Braune, K. F. Doerner, and R. F. Hartl, "A machine learning approach for flow shop scheduling problems with alternative resources, sequence-dependent setup times, and blocking," *OR Spectrum*, vol. 41, no. 4, pp. 871–893, 2019.

[14] Y. Hirochika and N. Hirofumi, "Estimation of processing time using machine learning and real factory data for optimization of parallel machine scheduling problem," *Operations Research Perspectives*, vol. 8, article 100196, 2021.

[15] X. Feng and Z. Xu, "Integrated production and transportation scheduling on parallel batch-processing machines," *IEEE Access*, vol. 7, pp. 148393–148400, 2019.

[16] J. F. Vijay, "Cloud data analysis using a genetic algorithm-based job scheduling process," *Expert Systems*, vol. 36, no. 5, 2019.

[17] L. S. Dias and M. G. Ierapetritou, "Integration of planning, scheduling and control problems using data-driven feasibility analysis and surrogate models," *Computers & Chemical Engineering*, vol. 134, p. 106714, 2020.

[18] A. Jlrd, A. Nf, and B. Tj, "Multi-process production scheduling with variable renewable integration and demand response," *European Journal of Operational Research*, vol. 281, no. 1, pp. 186–200, 2020.

[19] C. Tsay, A. Kumar, J. Flores-Cerrillo, and M. Baldea, "Optimal demand response scheduling of an industrial air separation unit using data-driven dynamic models," *Computers & Chemical Engineering*, vol. 126, pp. 22–34, 2019.

[20] O. J. Fisher, N. J. Watson, J. E. Escrig et al., "Considerations, challenges and opportunities when developing data-driven models for process manufacturing systems," *Computers & Chemical Engineering*, vol. 140, p. 106881, 2020.

[21] A. Allahverdi and F. S. al-Anzi, "A PSO and a Tabu search heuristics for the assembly scheduling problem of the two-stage distributed database application," *Computers and Operations Research*, vol. 33, no. 4, pp. 1056–1080, 2006.

[22] M. Yu, Y. Zhang, K. Chen, and D. Zhang, "Integration of process planning and scheduling using a hybrid GA/PSO algorithm," *International Journal of Advanced Manufacturing Technology*, vol. 78, no. 1-4, pp. 583–592, 2015.

[23] A. Brusaferri, E. Leo, L. Nicolosi, D. Ramin, and S. Spinelli, "Integrated automation system with PSO based scheduling for PCB remanufacturing plants," in *2019 IEEE 17th International Conference on Industrial Informatics (INDIN)*, Helsinki, Finland, 2019.

[24] Y. Yin, S. Linfu, and Y. Chengfeng, "Particle swarm optimization algorithm for real-time parameter evaluation based on embedded diamond thinking," *Journal of Systems Simulation*, vol. 21, no. 14, pp. 4317–4323, 2009.

[25] H. Kaijun and L. Huaiwei, "DPSO-based task scheduling algorithm in cloud environment," *Computer Engineering*, vol. 40, no. 1, pp. 59–62, 2014.

[26] Y. Chengyu, C. Dongning, and Z. Ruixing, "Hybrid-particle interaction particle swarm optimization algorithm," *Chinese Journal of Mechanical Engineering*, vol. 51, no. 6, pp. 198–207, 2015.

Research Article

Position Estimation and Error Correction of Mobile Robots Based on UWB and Multisensors

Jie Li [ID],[1] Jiameng Xue [ID],[2] Di Fu [ID],[1] Chao Gui [ID],[1] and Xingsong Wang [ID][1]

[1]*School of Mechanical Engineering, Southeast University, Nanjing, Jiangsu 211189, China*
[2]*School of Engineering, The University of Manchester, Manchester M13 9PL, UK*

Correspondence should be addressed to Xingsong Wang; xswang@seu.edu.cn

Since there are many interferences in the indoor environment, it is difficult to achieve the precise positioning of the mobile robot using a single sensor. This paper presents a position estimation and positioning error correction method of mobile robots based on multisensor data. The robot's positioning sensor includes ultra-wideband (UWB) components, inertial measurement unit (IMU), and encoders. UWB multipath interference causes more ranging errors, which can be reduced by the correction equation after data fitting. The real-time coordinates of the UWB robot tag can be calculated based on multiple UWB anchor data and the least squares method. The coordinate data (x_c, y_c) are acquired by UWB positioning subsystem, and the velocity data $(\dot{x}_c, \dot{y}_c)$ are collected by IMU together with encoders. The multisensor data continuously update Kalman filter and estimate robot position. In the positioning process, the positioning data of different sensors can be mutually corrected and supplemented. The results of UWB ranging correction experiments indicate that data fitting can improve the UWB positioning accuracy. In the multisensor positioning experiments, compared with a single sensor, the positioning method based on data fusion of UWB, IMU, and encoders has higher accuracy and adaptability. When UWB signals are interfered or invalid, other sensors can still work normally and complete the robot positioning process. The multisensor positioning method not only improves the robot positioning accuracy but also has stronger environmental adaptability.

1. Introduction

Various positioning methods and sensors have been developed and applied to different fields, such as underground garages, smart factories, airports, and restaurants. In outdoor positioning, Global Positioning System (GPS) is an important positioning method, which can perform ranging and positioning of cars, ships, and mobile phones through satellites. However, due to the interference of obstacles (such as roofs, walls, and furniture) in the indoor environment, it is more difficult to achieve high-precision positioning. Positioning of indoor mobile robots has also become one of the research hotpots in recent years. Using different positioning sensors, the robot can obtain more accurate position information. Accurate positioning and navigation can improve the automation and intelligence of mobile robots.

There are many indoor positioning methods with different sensors, such as Radio-Frequency Identification (RFID)

positioning [1], ultrasonic positioning, Bluetooth positioning, wireless sensor networks (WSN) [2, 3], UWB positioning [4], Inertial Navigation System (INS), and Concurrent Mapping and Localization (CML/SLAM). The combination of ranging sensors is beneficial to improve the positioning and navigation accuracy of mobile robots, for example, a localization and pose estimation method for mobile robot navigation using passive RFID [5], a lightweight indoor robot positioning system which operated on cost-effective WiFi-based received signal strength (RSS) [6], a novel accurate hybrid global self-localization algorithm for mobile robots based on ultrasonic localization system [7], and an efficient indoor localization algorithm based on a light detection and ranging (LiDAR) device. The UWB positioning method has been gradually applied due to the advantages of long positioning distance and high stability [8–11].

Compared with other positioning methods, UWB positioning devices can be installed and implemented in a short
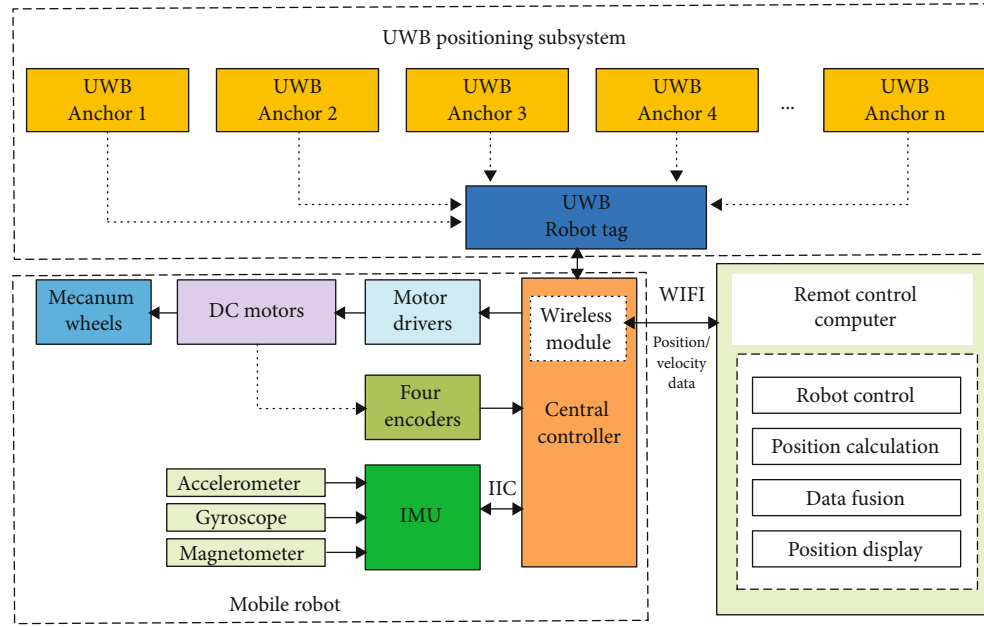
FIGURE 1: Robot positioning system.
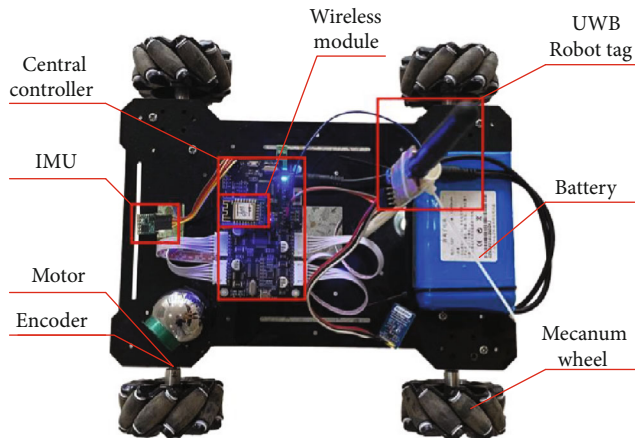
FIGURE 2: Mecanum wheeled mobile robot and positioning sensors.

time. UWB positioning technology utilizes distributed sensor devices for distance estimation. The distances between multiple UWB anchors and the UWB robot tag are measured simultaneously to estimate the robot position. The positioning methods include the Angle of Arrival (AOA) algorithm [12], the received signal strength (RSS) algorithm [13], the Time of Arrival (TOA) algorithm [14], the Time Difference of Arrival (TDOA) algorithm [15], and other hybrid algorithms [16–18]. The stability and accuracy of measured data are critical to the robot positioning. However, in the indoor environment, obstacle interference may reduce UWB positioning accuracy and even cause UWB positioning failure. Usually, increasing the number of UWB anchors can expand its positioning range, but it will also bring more errors and cause instability. Therefore, the error correction of UWB positioning data is necessary to improve its positioning accuracy.

It is difficult for a single sensor to achieve high accuracy due to sensor error or instability. The combined applications of multiple sensors are beneficial to improve positioning accuracy and stability. A method to accurately locate persons indoors by fusing INS with active RFID was presented in [19]. A method to integrate LiDAR and IMU was proposed to overcome the problem of the low accuracy and large accumulated errors of indoor mobile navigation and positioning [20]. In order to improve the accuracy of the data fusion filter, a tightly coupled UWB/INS-integrated scheme for indoor human navigation was investigated [21]. UWB has also been combined with other sensors for data fusion to achieve more precise positioning [22, 23].

Due to the interference of obstacles in the indoor environment, data fusion and filtering of UWB and multisensors are developed to improve the positioning accuracy of mobile robots. This paper proposed a position estimation and error correction method based on multisensor data. The robot positioning system can simultaneously acquire and integrate the data from UWB, IMU, and encoders. The double-sided two way ranging (DS-TWR) and data fitting method is used to improve the UWB ranging accuracy and correct errors caused by multipath signals. Meanwhile, the robot position coordinates were estimated by UWB anchors and the robot UWB robot tag. In the process of multisensor robot positioning, the heading angle error of IMU can be corrected by UWB positioning data. The velocity data calculated by IMU together with encoders and the UWB positioning data continuously update Kalman filter to estimate the robot position. The multisensor positioning method is conducive to improve the robot positioning accuracy. Even if UWB signals are interfered or shielded, the robot positioning can still be executed by other sensor data, which improves the practicability and anti-interference ability of the robot positioning system.
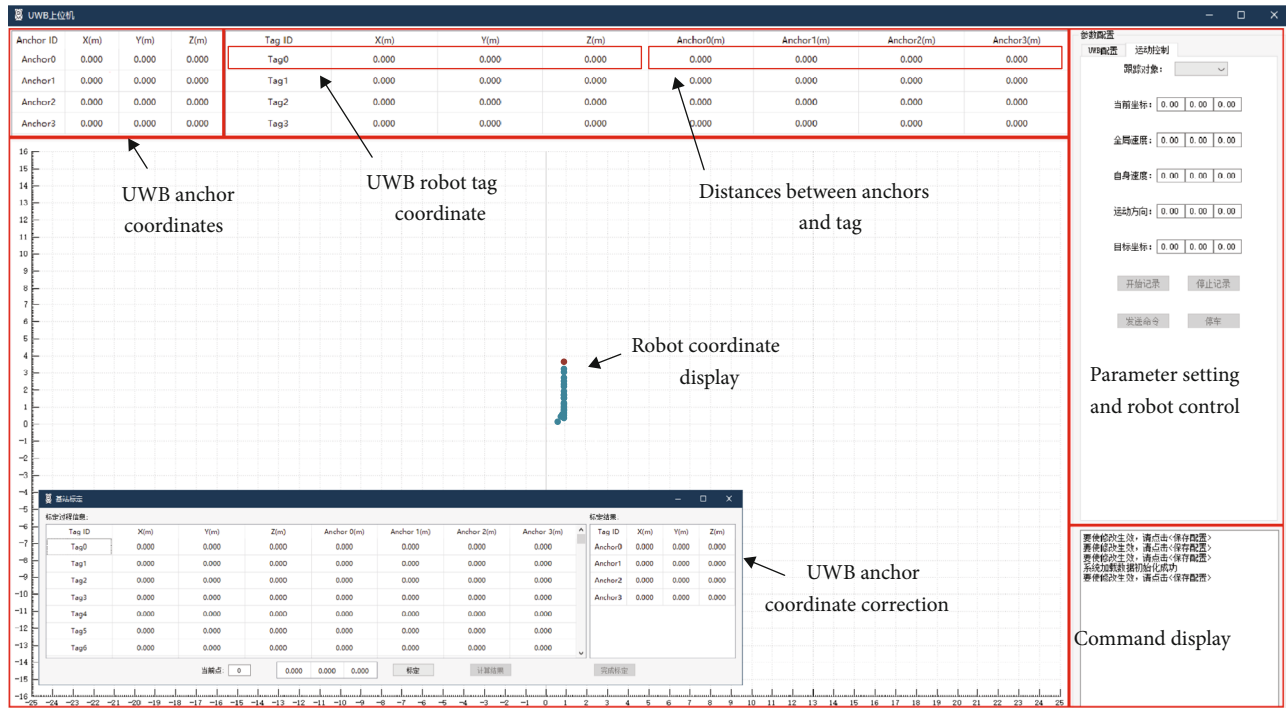
FIGURE 3: Software interface of the robot positioning system.

2. Positioning System Design

2.1. Positioning System Architecture. In a complex and multi-interference environment, multisensor cooperation has the ability of improving positioning accuracy. As shown in Figure 1, the robot positioning system consists of a UWB positioning subsystem, a remote control computer, and a mobile robot. The UWB positioning subsystem includes a UWB robot tag and some UWB anchors, which are fixedly installed in the surrounding environment. Based on the measured distance between the UWB robot tag and UWB anchors, the remote control computer executes positioning algorithms to calculate the position and coordinate of the robot.

The remote control computer communicates with the robot through the wireless module. It is used to process positioning data, calculate robot's coordinate, and control the mobile robot. The main functions include interactive communication, robot control, position estimation, data fusion and processing, and robot position display.

A mecanum wheeled mobile robot is considered in this work. As shown in Figure 2, the mobile robot consists of a central controller, four DC motors, four drivers, four encoders, and four mecanum wheels. The encoders are used to measure the rotational speed of each wheel and calculate the robot's velocity in self coordinate system based on the kinematic equations. An IMU sensor, composed of a three-axis accelerometer, a three-axis gyroscope, and a three-axis magnetometer, is installed on the robot. This sensor is used to estimate the heading angle of the robot during driving.

In terms of sensors that we used in the robot positioning system, the positioning data (i.e., the robot's position and velocity) are obtained by three different sensors: UWB,

encoders, and IMU. Each sensor inevitably has some errors during the positioning process, which may contain one or more of the following:

(1) During the running process, if the robot wheel slips, their rotational speeds measured by the encoders will have some errors. The heading angle and mileage of the robot integrated by velocities also have cumulative errors, which will increase over time

(2) The robot's heading angle calculated by the IMU has less error in a short operating time. However, if the correction is not carried out for a long time, larger cumulative angle errors will occur

(3) UWB positioning is a direct positioning method; the positioning error will not accumulate over time. However, the accuracy and stability of UWB positioning are not as good as expected. Besides, UWB positioning may be invalid due to the interference of obstacles in the environment

As discussed above, different positioning sensors have their own errors and shortcomings; we proposed a multisensor data fusion algorithm. By fusing the data of UWB, IMU, and encoders, the algorithm can effectively improve positioning accuracy, reduce cumulative errors, and eliminate obstacle interference. The positioning data obtained by the sensors are mutually corrected and supplemented. On the one hand, encoders and IMU have higher accuracy and freedom from obstacles in a short time but have larger cumulative errors. The cumulative angle errors can be reduced by using UWB positioning data, which will correct the heading angle calculated by the IMU. On the other hand, due to
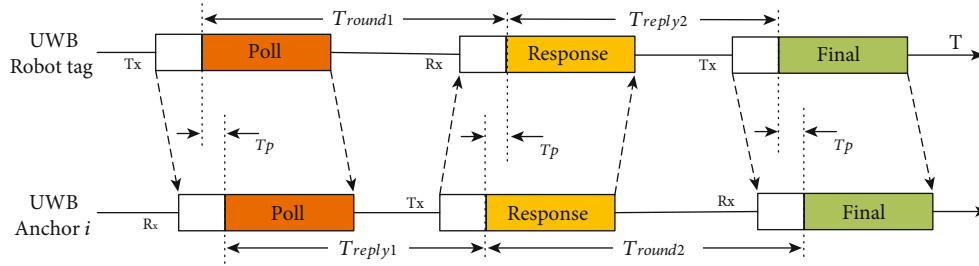
FIGURE 4: UWB signal flight time diagram based on the DS-TWR method.
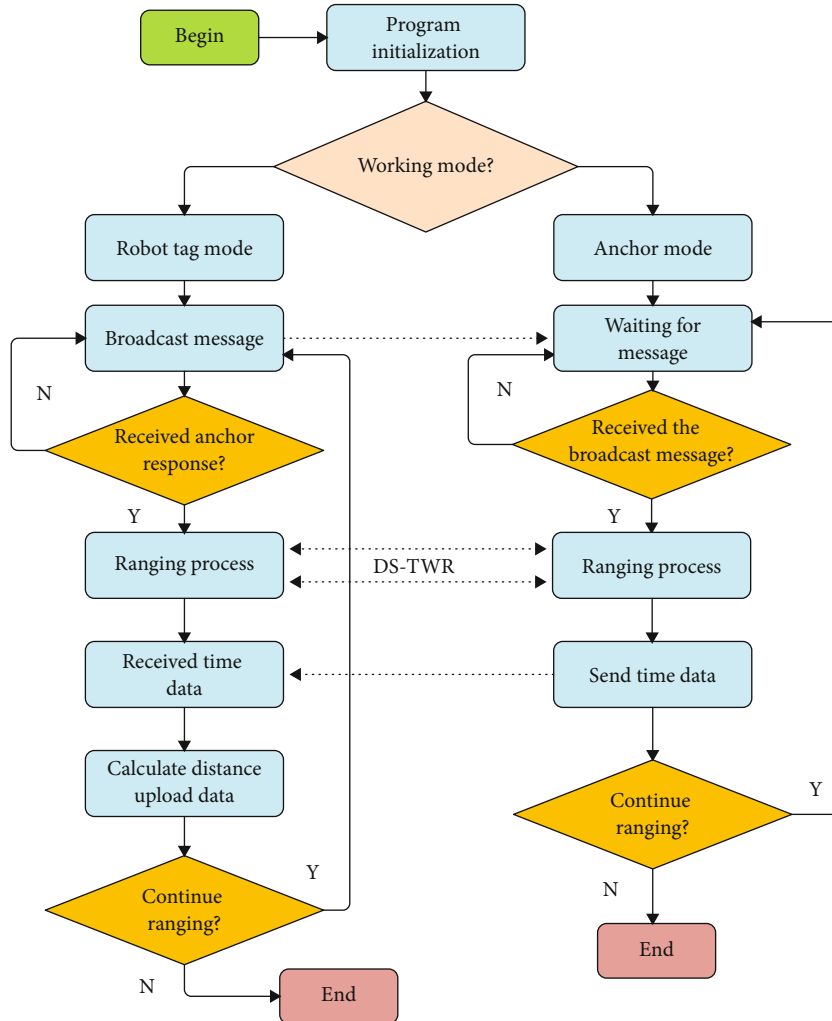
FIGURE 5: Flowchart of the UWB robot tag and UWB anchors.

environmental interference, the positioning error of UWB may become larger. The UWB's positioning error can be corrected through the IMU and encoder data as well, which will improve the robot positioning accuracy.

2.2. Software Development. As shown in Figure 3, we developed a software interface of the robot system. Visualized software is developed on the remote control computer, and

the main functions include coordinate display of UWB anchors, real-time coordinate and trajectory display of the robot, robot control, and data processing.

3. UWB Positioning and Error Correction

3.1. UWB Ranging Method. The UWB positioning subsystem performs positioning and coordinate calculation by
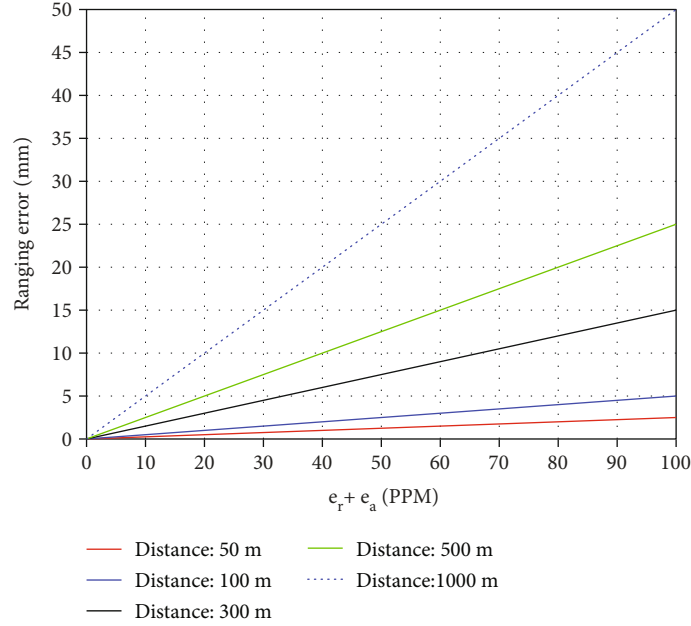
FIGURE 6: UWB ranging errors at different clock skews.

measuring the distances between the UWB tag and multiple UWB anchors. Time of Fly (TOF) method is used to measure the UWB signal flight distance; the formula is

$$\Delta d = C_u \cdot \Delta t, \tag{1}$$

where C_u is the UWB signal transmission speed, which approximately equals to the speed of light, and Δt is the time difference of signal flight.

However, due to the inability to accurately synchronize the clock between the UWB robot tag and UWB anchors, the clock error will cause a large measurement error. For example, 1 ns clock deviation can cause a 30 cm ranging error. The DS-TWR method [24] was proposed to reduce the ranging error caused by clock unsynchronization and clock skews. As shown in Figure 4, the DS-TWR method can achieve more accurate UWB ranging by measuring the three round-trip times between the UWB robot tag and the UWB anchors. Figure 5 shows the working flowchart of the UWB robot tag and UWB anchors.

The UWB signal flight time T_p between the UWB tag and the UWB anchor is

$$T_p = \frac{\left(T_{\text{round1}} \times T_{\text{round2}} - T_{\text{reply1}} \times T_{\text{reply2}}\right)}{\left(T_{\text{round1}} + T_{\text{round2}} + T_{\text{reply1}} + T_{\text{reply2}}\right)}. \tag{2}$$

Because the signal flight time T_p is much shorter than the processing data time T_{reply1} and T_{reply2}, the flight time of the signal after adding the clock errors approximately equals to

$$\widehat{T}_p \approx \frac{2 \cdot (1 + e_r)(1 + e_a)}{(1 + e_r) + (1 + e_a)} T_p, \tag{3}$$

where e_r and e_a are the clock skews of the UWB tag module and UWB anchor module, respectively.

As a result, the ranging error by the DS-TWR method is

$$T_e = \widehat{T}_p - T_p = \frac{e_r + e_a + e_r e_a}{2(1 + e_r)(1 + e_a)} \widehat{T}_p \approx \frac{e_r + e_a}{2} \widehat{T}_p. \tag{4}$$

Figure 6 shows the DS-TWR ranging error curve between the UWB robot tag and the UWB anchors at different distances and different clock skews. The results indicate that the ranging errors caused by clock skews are much smaller than the UWB measurement accuracy (about 100 mm). When the measuring distance is 300 m and the clock skew is 20 PPM, the ranging error is about 6 mm. During the robot actual positioning process, the UWB module with 5 PPM clock was adopted to improve the measurement accuracy.

3.2. Multipath Error Correction. As shown in Figure 7(a), in a line-of-sight (LOS) environment [25], UWB signals can transmit in a straight line without any obstruction. However, multipath interference may exist due to the reflection of walls and objects. Signals often transmit through multiple paths and cause some ranging errors. As shown in Figure 7(b), paths A and B are direct paths, and paths C and D are multipath interference; their length relationship is $D \approx C > B \approx A$. Although UWB has a strong ability to resist multipath interference, it is difficult to guarantee its ranging accuracy due to its ultrawide bandwidth and interference of complex indoor reflection environment.

The ranging error caused by multipath is related to the real environment. The method of data fitting is proposed to correct the error and improve the positioning accuracy.

(a)   (b)
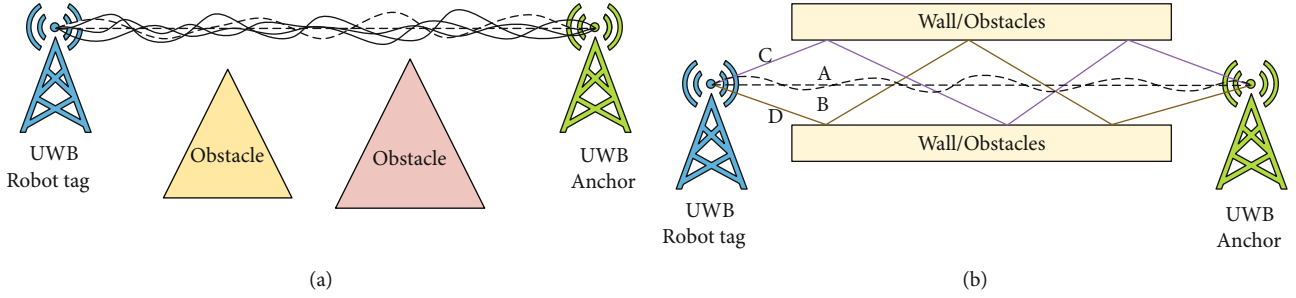
FIGURE 7: Broadcast environment analysis of UWB signals: (a) LOS environment without obstructions; (b) UWB multipath interference.
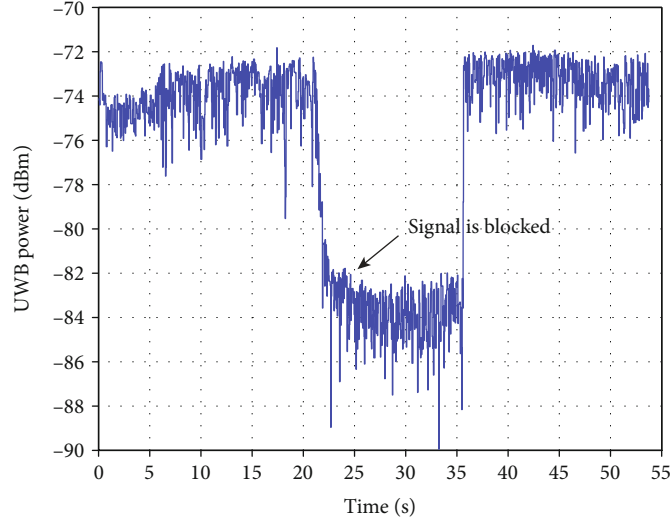
FIGURE 8: First path power of UWB signals under occlusion.

Assume that the relationship equation between the real distance and the UWB measurement distance is

$$d_t = a_0 + a_1 d_m + a_2 d_m^2 + \cdots + a_k d_m^k, \qquad (5)$$

where d_t is the real distance value and d_m is the UWB measured value. By measuring multiple sets of UWB data (d_{mi}, d_{ti}) in the LOS environment, we can get

$$d_{ti} = a_0 + a_1 d_{mi} + a_2 d_{mi}^2 + \cdots + a_k d_{mi}^k, \quad i = 1, 2, \cdots, n. \quad (6)$$

Suppose $\mathbf{A}_c = [a_0 \ a_1 \ \cdots \ a_k]^T$ and $\mathbf{D}_t = [d_{t1} \ d_{t2} \ \cdots \ d_{tn}]^T$; the matrix form of Equation (6) is

$$\mathbf{D}_t = \mathbf{D}_m \mathbf{A}_c, \qquad (7)$$

where $\mathbf{D}_m$ is

$$\mathbf{D}_m = \begin{bmatrix} 1 & d_{m1} & \cdots & d_{m1}^k \\ 1 & d_{m2} & \cdots & d_{m2}^k \\ 1 & d_{m3} & \vdots & \vdots \\ 1 & d_{m4} & \cdots & d_{mn}^k \end{bmatrix}. \qquad (8)$$

Solving the matrix equation by the principle of least squares, we have

$$\mathbf{A}_c = \left(\mathbf{D}_m^T \mathbf{D}_m\right)^{-1} \mathbf{D}_m^T \mathbf{D}_t. \qquad (9)$$

By substituting the coefficient matrix $\mathbf{A}_c = [a_0 \ a_1 \ \cdots \ a_k]^T$ into Equation (5), the relationship between the UWB measured distance and the real distance can be calculated. Data fitting is conducive to correcting the errors caused by UWB multipath and improving the ranging accuracy. A large number of experiments we have conducted indicate that when the data fitting equation is $d_t = a_0 + a_1 d_m$, the ranging accuracy is high enough to satisfy the operating requirements.

3.3. First Path Power. Due to the interference of UWB signals, the ranging accuracy of UWB anchors cannot be guaranteed. It means that if severely disturbed ranging data is taken into account, the positioning accuracy of the mobile robot will be greatly reduced. To solve this problem, the first path power (or direct path power) of UWB signals is collected to determine the availability of
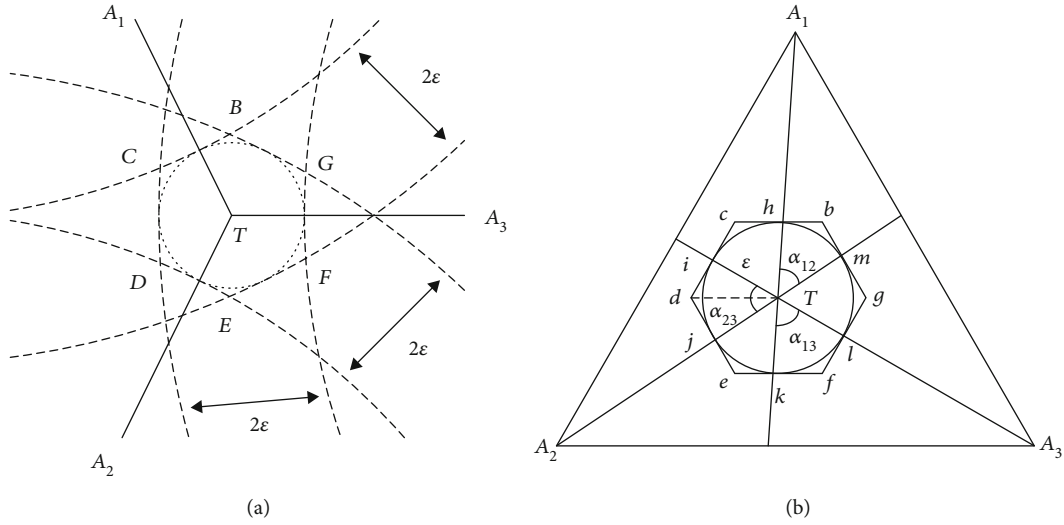
FIGURE 9: Geometrical relationship of UWB anchor layout: (a) measured distance error analysis; (b) layout analysis of UWB anchors.

UWB ranging data, which can be calculated as

$$P_F = 10 \times \log_{10}\left(\frac{F_1{}^2 + F_2{}^2 + F_3{}^2}{N_0{}^2}\right) - P_0(\mathrm{dBm}), \qquad (10)$$

where F_1, F_2, F_3, and N_0 are internal register variables of the UWB module and P_0 is a constant.

The first path power can reflect the signal propagation state between the UWB robot tag and UWB anchors. When the UWB signal is blocked, the first path power will decrease significantly. Figure 8 shows the first path power of the UWB signal under occlusion. In a noninterference environment, the first path power of the UWB signal is usually maintained between -76 dBm and -72 dBm. When the signal is blocked by obstacles, the first path power will drop dramatically to -82 dBm or lower.

By analyzing the first path power strength of UWB signals, the data availability of UWB anchors can be confirmed. For accurate positioning, the ranging data with power below a certain value will be discarded. It is considered that there are obstacles between the UWB robot tag and the UWB anchor; thus, this UWB anchor is invalid. If the number of available UWB anchors cannot satisfy the requirement of the robot positioning, UWB positioning will fail and the robot coordinate cannot be calculated.

3.4. UWB Anchor Layout Analysis. In the UWB positioning subsystem, multiple UWB anchor data are used simultaneously to calculate the robot position. The layout of UWB anchors has a significant impact on the robot positioning accuracy. For high-precision robot positioning, the layout and installation positions of the UWB anchors should meet the following principles:

(1) The operating area of the robot should be covered by at least 3 UWB anchor signals

(2) The signal coverage of UWB anchors should be fully utilized

(3) The horizontal dilution of precision (HDOP) of UWB positioning should be small enough and UWB anchors should avoid being clustered in a small area

If the real distance between UWB anchors and the UWB robot tag is $d_i^{(r)}$, $(i = 1, 2, 3 \cdots)$ and the UWB ranging error is $\varepsilon(\varepsilon \geq 0)$, the measured distance can be expressed as

$$d_i^{(m)} \in \left[d_i^{(r)} - \varepsilon, d_i^{(r)} + \varepsilon\right] \quad (i = 1, 2, 3 \cdots). \qquad (11)$$

As shown in Figure 9(a), assuming that three UWB anchors are located at points $A1$, $A2$, and $A3$, their measured distances are $d_1^{(m)}$, $d_2^{(m)}$, and $d_3^{(m)}$. The three circles, with points $A1$, $A2$, and $A3$ as the centers and $d_1^{(m)}$, $d_2^{(m)}$, and $d_3^{(m)}$ as the radii, intersect to form Zone $BCDEFG$, which is the possible coordinate range of the robot. Its area S_{BCDEFG} represents the robot positioning error.

Because ε is small, Zone $BCDEFG$ is approximately hexagon $bcdefg$ in Figure 9(b), which is the circumscribed hexagon of circle T. The relative angles of three UWB anchors are α_{12}, α_{23}, and α_{13}, respectively. According to its symmetry,

$$\begin{cases} S_{Tmbh} \approx S_{Tjek} \\ S_{Tidj} \approx S_{Tlgm} \\ S_{Tkfl} \approx S_{Thci} \end{cases} \qquad (12)$$

The area of triangle Tid is

$$S_{Tid} = \frac{\varepsilon^2}{2} \tan\left(\frac{\alpha_{23}}{2}\right). \qquad (13)$$
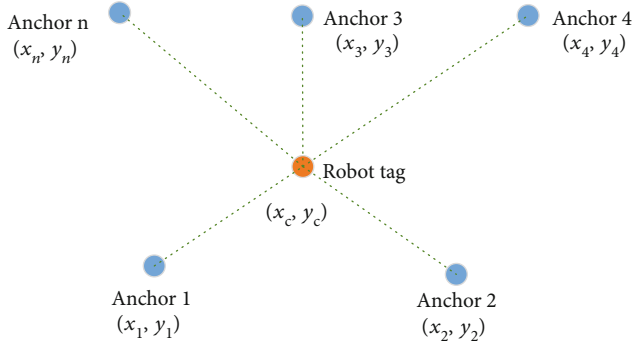
FIGURE 10: UWB positioning and coordinate calculation.
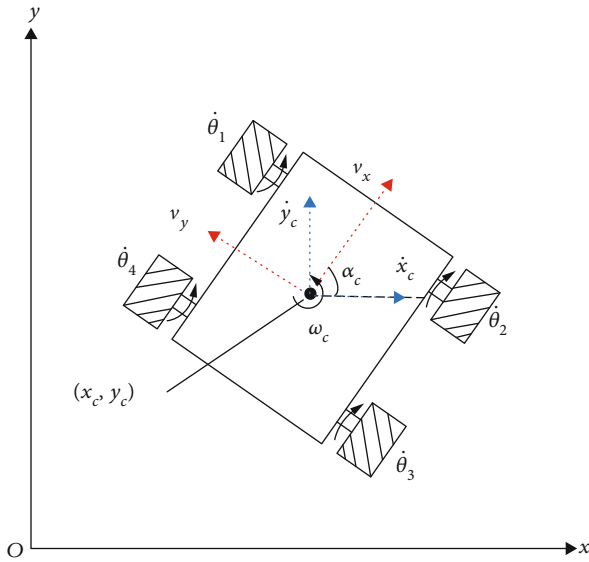
FIGURE 11: Robot velocity calculation in the absolute coordinate system.

Therefore, we have

$$S_{Tidj} = \varepsilon^2 \tan\left(\frac{\alpha_{23}}{2}\right). \tag{14}$$

Similarly,

$$S_{Tmbh} = \varepsilon^2 \tan\left(\frac{\alpha_{12}}{2}\right), S_{Tkfl} = \varepsilon^2 \tan\left(\frac{\alpha_{13}}{2}\right). \tag{15}$$

The area of hexagon $bcdefg$ is

$$S_{bcdefg} = 2\varepsilon^2 \left(\tan\left(\frac{\alpha_{13}}{2}\right) + \tan\left(\frac{\alpha_{12}}{2}\right) + \tan\left(\frac{\alpha_{23}}{2}\right)\right), (\alpha_{13} + \alpha_{12} + \alpha_{23} = \pi). \tag{16}$$

When $\alpha_{13} = \alpha_{12} = \alpha_{23}$, S_{bcdefg} has the smallest value:

$$S_{bcdefg} = 6\varepsilon^2 \frac{1}{3}\left(\tan\left(\frac{\alpha_{13}}{2}\right) + \tan\left(\frac{\alpha_{12}}{2}\right) + \tan\left(\frac{\alpha_{23}}{2}\right)\right) \geq 6\varepsilon^2 \tan\left(\frac{\pi}{6}\right). \tag{17}$$

When three UWB anchors are used for robot positioning, the equilateral triangle layout is conducive to reducing positioning errors. Similarly, when the number of UWB anchors is four, the square layout is more favorable for the positioning.

4. Robot Position Estimation

4.1. UWB Coordinate Estimation. The UWB positioning subsystem includes UWB anchors and a UWB robot tag. UWB anchors are generally fixed at certain positions with known coordinates, and the UWB robot tag is installed on the robot. As shown in Figure 10, there are some UWB anchors with known coordinates (x_i, y_i) on a 2D plane: Anchor 1, Anchor 2, Anchor 3, ..., Anchor n, and the coordinate of the UWB robot tag is (x_c, y_c). The distances from the UWB robot tag to the UWB anchors measured by the robot positioning system are $d_i (i = 1, 2, \cdots, n)$. From the geometric relationship, we can get

$$\begin{cases} (x_c - x_1)^2 + (y_c - y_1)^2 = d_1^2 \\ (x_c - x_2)^2 + (y_c - y_2)^2 = d_2^2 \\ (x_c - x_3)^2 + (y_c - y_3)^2 = d_3^2 \\ \cdots \\ (x_c - x_n)^2 + (y_c - y_n)^2 = d_n^2 \end{cases} \tag{18}$$

The following matrix equation can be obtained by Equation (18):

$$\mathbf{A}_u \mathbf{X}_u = \mathbf{D}_u, \tag{19}$$

where,

$$\mathbf{A}_u = \begin{pmatrix} 2(x_1 - x_n) & 2(y_1 - y_n) \\ 2(x_2 - x_n) & 2(y_2 - y_n) \\ 2(x_3 - x_n) & 2(y_3 - y_n) \\ \vdots \\ 2(x_{n-1} - x_n) & 2(y_{n-1} - y_n) \end{pmatrix},$$

$$\mathbf{D}_u = \begin{pmatrix} x_1^2 - x_n^2 + y_1^2 - y_n^2 + d_n^2 - d_1^2 \\ x_2^2 - x_n^2 + y_2^2 - y_n^2 + d_n^2 - d_2^2 \\ x_3^2 - x_n^2 + y_3^2 - y_n^2 + d_n^2 - d_3^2 \\ \vdots \\ x_{n-1}^2 - x_n^2 + y_{n-1}^2 - y_n^2 + d_n^2 - d_{n-1}^2 \end{pmatrix}, \mathbf{X}_u = \begin{bmatrix} x_c \\ y_c \end{bmatrix}. \tag{20}$$

Because the UWB positioning subsystem has some positioning error, equation (19) can be corrected as

$$\mathbf{A}_u \mathbf{X}_u + \mathbf{N} = \mathbf{D}_u, \tag{21}$$

where $\mathbf{N}$ is an error vector with $n - 1$ dimensions.
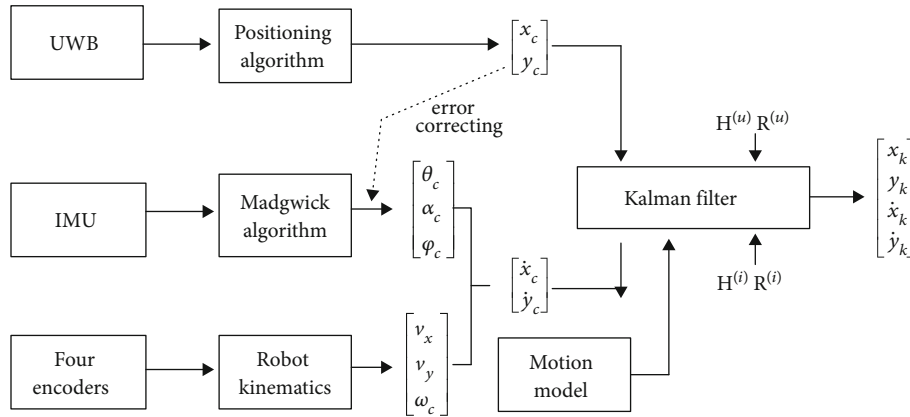
FIGURE 12: Multisensor data fusion and robot positioning estimation.

(a)

(b)
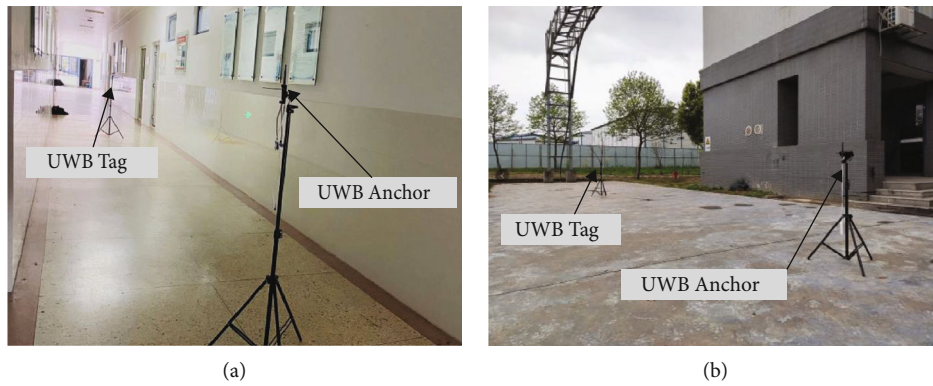
FIGURE 13: UWB ranging correction experiments: (a) indoor correction experiment; (b) outdoor correction experiment.

When $\mathbf{N} = \mathbf{D}_u - \mathbf{A}_u\mathbf{X}_u$ is the smallest, the measurement error will be the smallest. Suppose $f(x) = \mathbf{D}_u - \mathbf{A}_u\mathbf{X}_u$, the least square estimate of $\mathbf{X}$ is

$$\mathbf{X}_u = \left(\mathbf{A}_u{}^T\mathbf{A}_u\right)^{-1}\mathbf{A}_u{}^T\mathbf{D}_u. \tag{22}$$

Therefore, (x_c, y_c) is the estimated coordinate value of the robot. When positioning the mobile robot in the plane,

at least three UWB anchors are required. Generally, more UWB anchors can improve the robot positioning accuracy and stability. However, too many UWB anchor data may cause more calculations and other possible errors.

4.2. Robot Velocity Calculation.
In the robot positioning process, the robot's absolute coordinate (x_c, y_c) can be calculated by the UWB positioning subsystem. Meanwhile, the rotation speeds of the wheels are recorded by four encoders.

$$\begin{bmatrix} v_x \\ v_y \\ \omega_c \end{bmatrix} = \frac{R}{4} \cdot \begin{bmatrix} 1 & 1 & 1 & 1 \\ \tan\beta & -\tan\beta & \tan\beta & -\tan\beta \\ \dfrac{1}{W+\cot\beta\cdot L} & -\dfrac{1}{W+\cot\beta\cdot L} & -\dfrac{1}{W+\cot\beta\cdot L} & \dfrac{1}{W+\cot\beta\cdot L} \end{bmatrix} \cdot \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \\ \dot{\theta}_4 \end{bmatrix}, \tag{23}$$

where $\dot{\theta}_i$ is the rotation speed of robot's wheels, β is the inclination angle of the inner rollers of the mecanum wheels, and $\beta = \pi/4$. W and L are the half of robot's width and length, respectively. The running velocities of the robot are $(v_x, v_y,$

According to the robot kinematics, we can get $\omega_c)$, which represent the forward velocity, transverse velocity, and rotation velocity, respectively.

Based on the Madgwick algorithm [26], the IMU can calculate the three-axis angle of the robot $(\theta_c, \alpha_c, \varphi_c)$, which,

TABLE 1: Measured distance results of UWB indoors and outdoors.

Num	Actual distance (m)	Measured distance indoor (m)	Measured distance outdoor (m)
1	1.00	0.62	0.83
2	2.00	1.68	1.87
3	3.00	2.71	2.87
4	4.00	3.71	3.89
5	5.00	4.75	4.90
6	6.00	5.83	5.90
7	7.00	6.75	6.90
8	8.00	7.83	7.92
9	9.00	8.85	8.94
10	10.00	9.77	9.95
11	11.00	10.8	10.97
12	12.00	11.85	11.97
13	13.00	12.91	13.00
14	14.00	13.89	13.98
15	15.00	14.88	15.00
16	16.00	15.96	16.01
17	17.00	16.94	17.01
18	18.00	17.94	18.03
19	19.00	18.92	19.02
20	20.00	19.94	20.04

respectively, represents the roll angle, heading angle, and tilt angle. Since these angles have cumulative errors over time, robot's absolute coordinates (x_c, y_c) acquired by the UWB positioning subsystem are used to correct them. During the positioning process, the UWB positioning subsystem records the robot's moving trajectory in real time at a frequency of 50 Hz. Within a certain period of time, the robot's coordinates $\{(x_i, y_i), i = 1, 2, \cdots, N\}$ are used to evaluate linearity of the moving trajectory. When the linearity is greater than 0.9, the slope of the moving trajectory is used to correct the IMU heading angle. Otherwise, the robot moving trajectory is regarded as nonlinear, and the IMU heading angle will be not corrected.

Some errors may be caused by the sliding of robot's mecanum wheels, and the heading angle integrated by robot's rotation velocity ω_c may be inaccurate. As shown in Figure 11, the IMU heading angle α_c and the robot's running velocities (v_x, v_y) are combined to calculate robot velocities $(\dot{x}_c, \dot{y}_c)$ in the absolute coordinate system xOy:

$$\begin{cases} \dot{x}_c = v_x \cdot \cos \alpha_c - v_y \cdot \sin \alpha_c, \\ \dot{y}_c = v_x \cdot \sin \alpha_c + v_y \cdot \cos \alpha_c. \end{cases} \tag{24}$$

4.3. Multisensor Position Estimation.

As we mentioned above, environmental interference can cause UWB positioning errors. Therefore, a multisensor position estimation method is proposed to improve the positioning accuracy of the robot. As shown in Figure 12, the robot

multisensor data are fused and combined for more accurate position coordinate estimation. Robot positioning data are acquired by multiple sensors, including UWB positioning subsystem, IMU, and encoders.

As the accuracy of the UWB positioning data (x_c, y_c) is not high enough and $(\dot{x}_c, \dot{y}_c)$ has cumulative errors over time, Kalman filter is used to fuse multisensor data. Both (x_c, y_c) and $(\dot{x}_c, \dot{y}_c)$ are fed into the Kalman filter to estimate the robot coordinates and velocities $(x_k, y_k, \dot{x}_k, \dot{y}_k)$. The robot position estimation and prediction include two steps: first, combine the robot motion model and predict the robot state $(x_k, y_k, \dot{x}_k, \dot{y}_k)$ at time k based on the optimal estimation $(x_{k-1}, y_{k-1}, \dot{x}_{k-1}, \dot{y}_{k-1})$ at time $k - 1$; second, use the measured data (UWB positioning subsystem, IMU, and encoders) to correct the predicted state and its parameters.

During the positioning process, the robot receives coordinate data (x_c, y_c) and velocity data $(\dot{x}_c, \dot{y}_c)$ at frequencies of 50 Hz and 100 Hz, respectively. In each time period, assuming that the robot performs uniform motion, the state equation of the robot is

$$\mathbf{X}_k = \mathbf{\Phi} \cdot \mathbf{X}_{k-1} + \mathbf{\Gamma w}_{k-1}, \tag{24}$$

where $\mathbf{X}_k = \begin{bmatrix} x_k & \dot{x}_k & y_k & \dot{y}_k \end{bmatrix}^T$ is the state vector of the robot and $\mathbf{\Phi}$ is the state transition matrix of the uniform motion model.

$$\mathbf{\Phi} = \begin{bmatrix} 1 & dt^{(ui)} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & dt^{(ui)} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{25}$$

where $dt^{(ui)}$ is the sampling period of positioning data. Due to the different update frequencies of multisensor data, $dt^{(ui)}$ includes two cases: $dt^{(u)}$ is the sampling period of UWB positioning data (x_c, y_c), and $dt^{(i)}$ is the sampling period of velocity data $(\dot{x}_c, \dot{y}_c)$.

Suppose the noise of the robot motion system is $\mathbf{\Gamma w}_k$:

$$\mathbf{\Gamma} = \begin{bmatrix} dt^{(ui)} & 0 \\ 1 & 0 \\ 0 & dt^{(ui)} \\ 0 & 1 \end{bmatrix}, \tag{26}$$

$$\mathbf{w}_k = \begin{bmatrix} u_x \\ u_y \end{bmatrix},$$

where u_x and u_y are white Gaussian noise.

The measurement equation of the robot positioning system is

$$\mathbf{Z}_k = \mathbf{H}^{(ui)} \cdot \mathbf{X}_k + \mathbf{v}_k, \tag{27}$$

where $\mathbf{Z}_k = \begin{bmatrix} x_m & y_m \end{bmatrix}^T$ is the measurement vector, including UWB positioning data (x_c, y_c) and velocity data $(\dot{x}_c, \dot{y}_c)$. $\mathbf{v}_k$ is

(a)

(b)
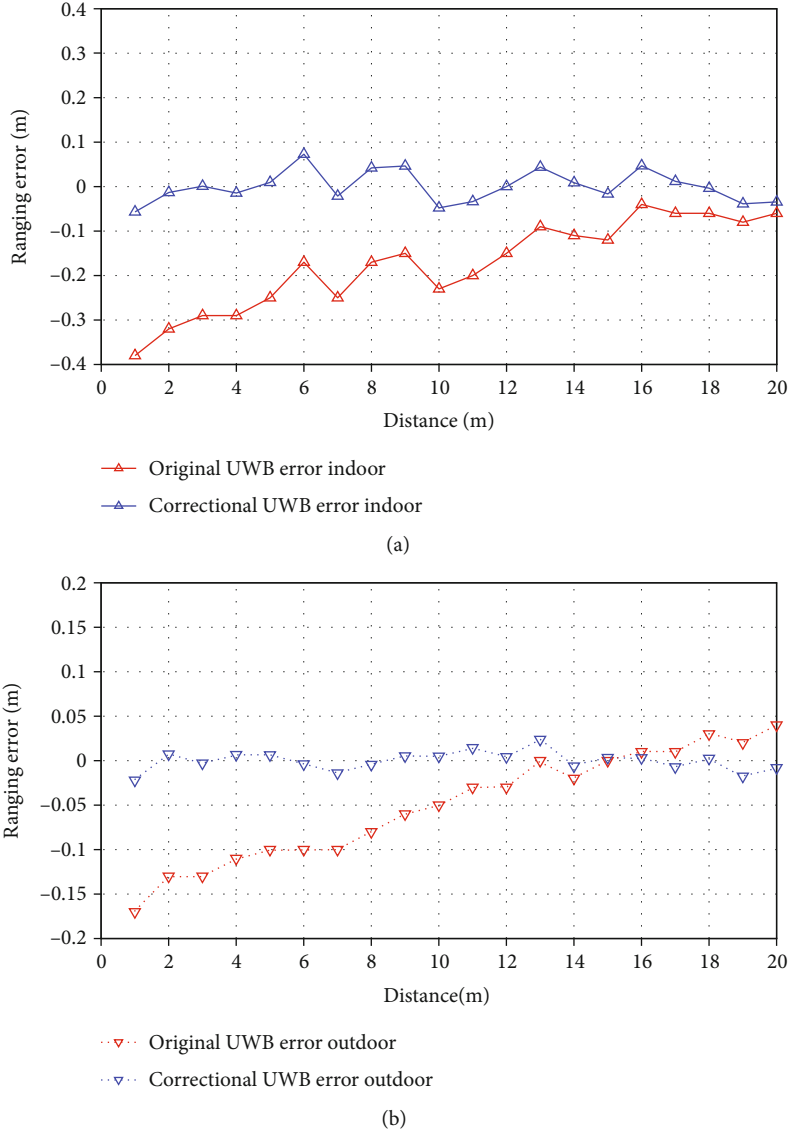
FIGURE 14: UWB ranging errors: (a) original ranging errors; (b) correctional ranging errors.

the measurement noise. $\mathbf{H}^{(ui)}$ represents the transfer relationship matrix from the state vector $\mathbf{X}_k$ to the measurement vector $\mathbf{Z}_k$. The observation matrices $\mathbf{H}^{(ui)}$ corresponding to (x_c, y_c) and $(\dot{x}_c, \dot{y}_c)$ are

$$\mathbf{H}^{(u)} = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \end{bmatrix},$$
$$\mathbf{H}^{(i)} = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 1 \end{bmatrix}. \tag{28}$$

Based on the state equation and measurement equation of the robot positioning system, Kalman filter prediction

equation can be constructed as follows:

$$\mathbf{X}_{(k|k-1)} = \mathbf{\Phi} \cdot \mathbf{X}_{(k-1|k-1)} + \mathbf{w}_{(k)}, \tag{29}$$

where $\mathbf{w}_{(k)}$ is noise, $\mathbf{X}_{(k-1|k-1)}$ is the optimal estimation vector at time $k - 1$, and $\mathbf{X}_{(k|k-1)}$ is the prediction vector at time k predicted by positioning data at time $k - 1$.

The covariance of the robot system at time k is

$$\mathbf{P}_{(k|k-1)} = \mathbf{\Phi} \cdot \mathbf{P}_{(k-1|k-1)} \cdot \mathbf{\Phi}^T + \mathbf{Q}, \tag{30}$$

where $\mathbf{P}_{(k-1|k-1)}$ is the covariance matrix of the robot system
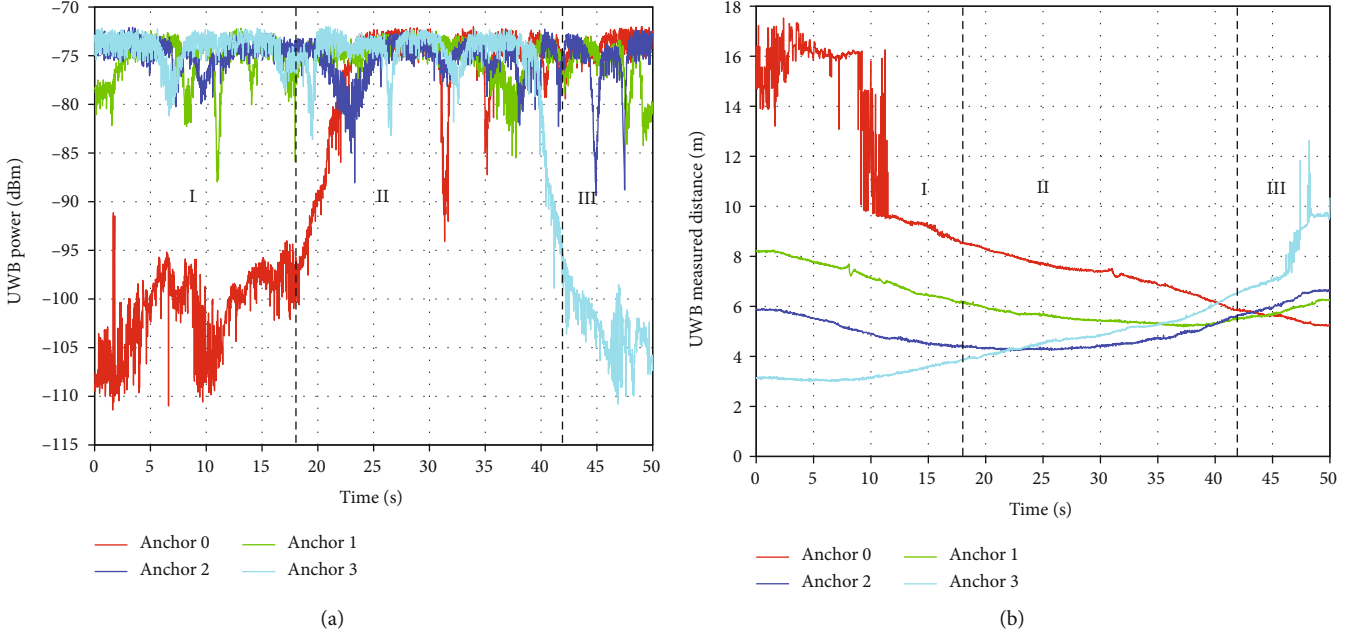
FIGURE 15: UWB ranging power and availability analysis: (a) the first path power of UWB anchors; (b) measured distances of UWB anchors.
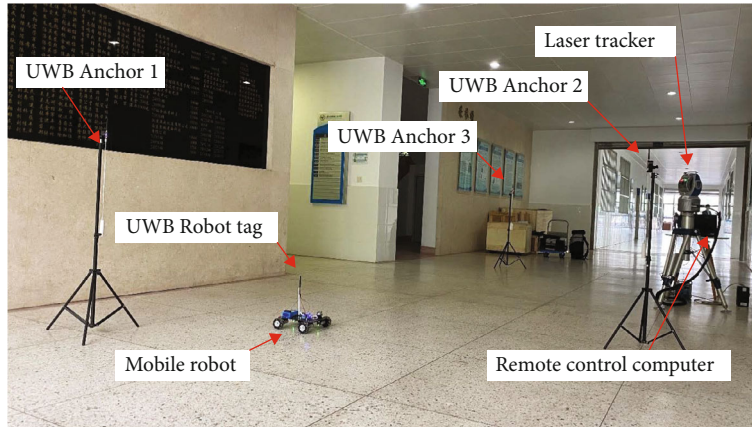
FIGURE 16: Multisensor positioning experiments.

at time $k-1$ and $\mathbf{Q}$ is the covariance of system noise:

$$\mathbf{Q} = \mathbf{G} \cdot \mathbf{G}^T \cdot K_e,$$

$$\mathbf{G} = \begin{bmatrix} \dfrac{e_t^2}{2} & 0 \\ e_t & 0 \\ 0 & \dfrac{e_t^2}{2} \\ 0 & e_t \end{bmatrix}. \qquad (31)$$

In the robot positioning process, $K_e = 0.25$ and $e_t = 0.02$.

The optimal estimation vector of the robot system at time k is

$$\mathbf{X}_{(k|k)} = \mathbf{X}_{(k|k-1)} + \mathbf{Kg}_{(k)}\left(\mathbf{Z}_{(k)} - \mathbf{H}^{(ui)}\mathbf{X}_{(k)}\right), \qquad (32)$$

where $\mathbf{Z}_{(k)}$ is the measured value, including two types of data: (x_c, y_c) and $(\dot{x}_c, \dot{y}_c)$. The $\mathbf{H}^{(ui)}$ corresponding to different measurement data is $\mathbf{H}^{(u)}$ and $\mathbf{H}^{(i)}$. Kalman filter gain $\mathbf{Kg}_{(k)}$ is

$$\mathbf{Kg}_{(k)} = \mathbf{P}_{(k|k-1)}\mathbf{H}^{(ui)^T}\left(\mathbf{H}^{(ui)}\mathbf{P}_{(k|k-1)}\mathbf{H}^{(ui)^T} + \mathbf{R}^{(ui)}\right)^{-1}, \qquad (33)$$

where $\mathbf{R}^{(ui)}$ is the covariance matrix of the observed noise. The value of $\mathbf{R}^{(ui)}$ represents the fluctuation of the observed data. Its value is larger, while the credibility of the data is lower, and its influence on the final estimation result is smaller; conversely, the smaller value of $\mathbf{R}^{(ui)}$ has a greater effect on the estimation result. The matrix $\mathbf{R}^{(ui)}$ corresponding to UWB positioning data (x_c, y_c) and velocity
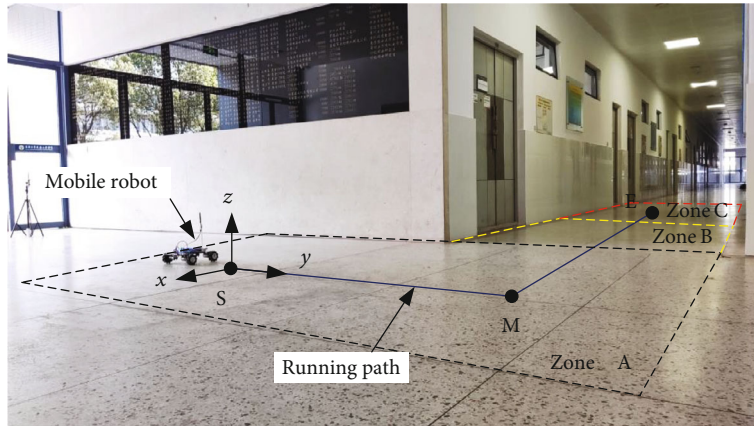
FIGURE 17: Experiment environment and robot moving trajectory.
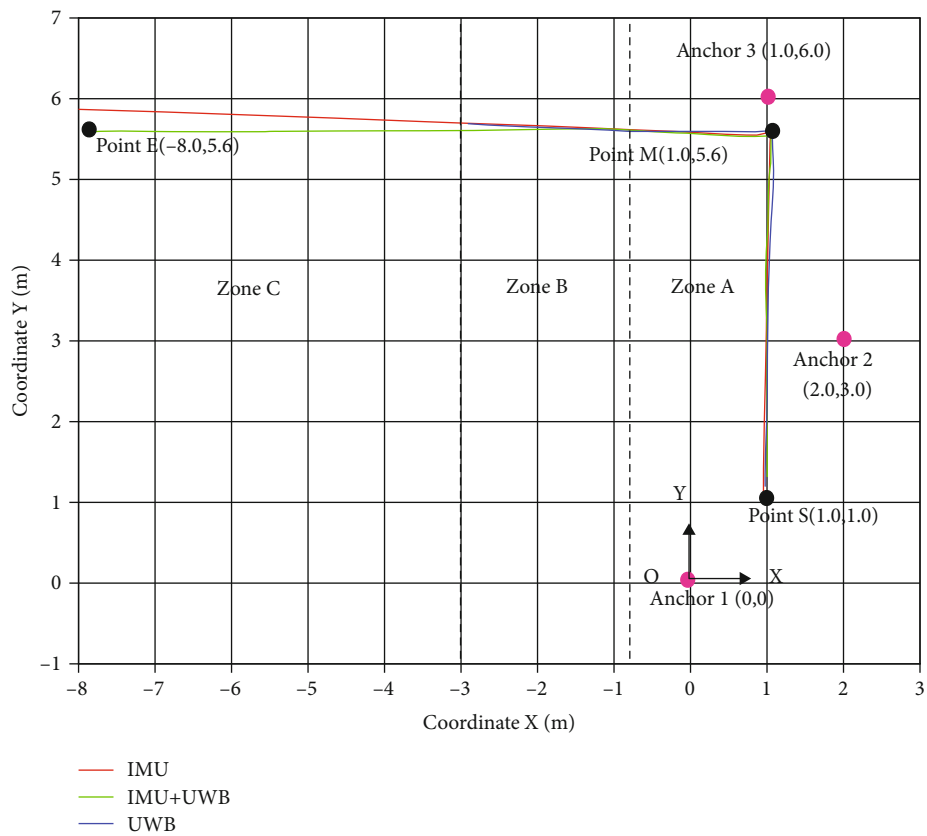
FIGURE 18: Coordinate system XOY and robot running trajectories by three positioning methods.

data$(\dot{x}_c, \dot{y}_c)$ is

$$\mathbf{R}^{(u)} = \begin{bmatrix} 0.09 & 0 \\ 0 & 0.09 \end{bmatrix},$$
$$\mathbf{R}^{(i)} = \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}. \tag{34}$$

Covariance matrix $\mathbf{P}_{(k|k)}$ of the robot system at time k is

updated to

$$\mathbf{P}_{(k|k)} = \left(\mathbf{I} - \mathbf{Kg}_{(k)}\mathbf{H}^{(ui)}\right)\mathbf{P}_{(k|k-1)}. \tag{35}$$

The optimal estimation $(x_k, \dot{x}_k, y_k, \dot{y}_k)$ of the robot position at time k can be calculated from Equation (32). Based on the optimal estimate and the covariance matrix at time k, combined with the measured data at time $k + 1$, the optimal estimate $(x_{k+1}, \dot{x}_{k+1}, y_{k+1}, \dot{y}_{k+1})$ at time $k + 1$ can be predicted successively.
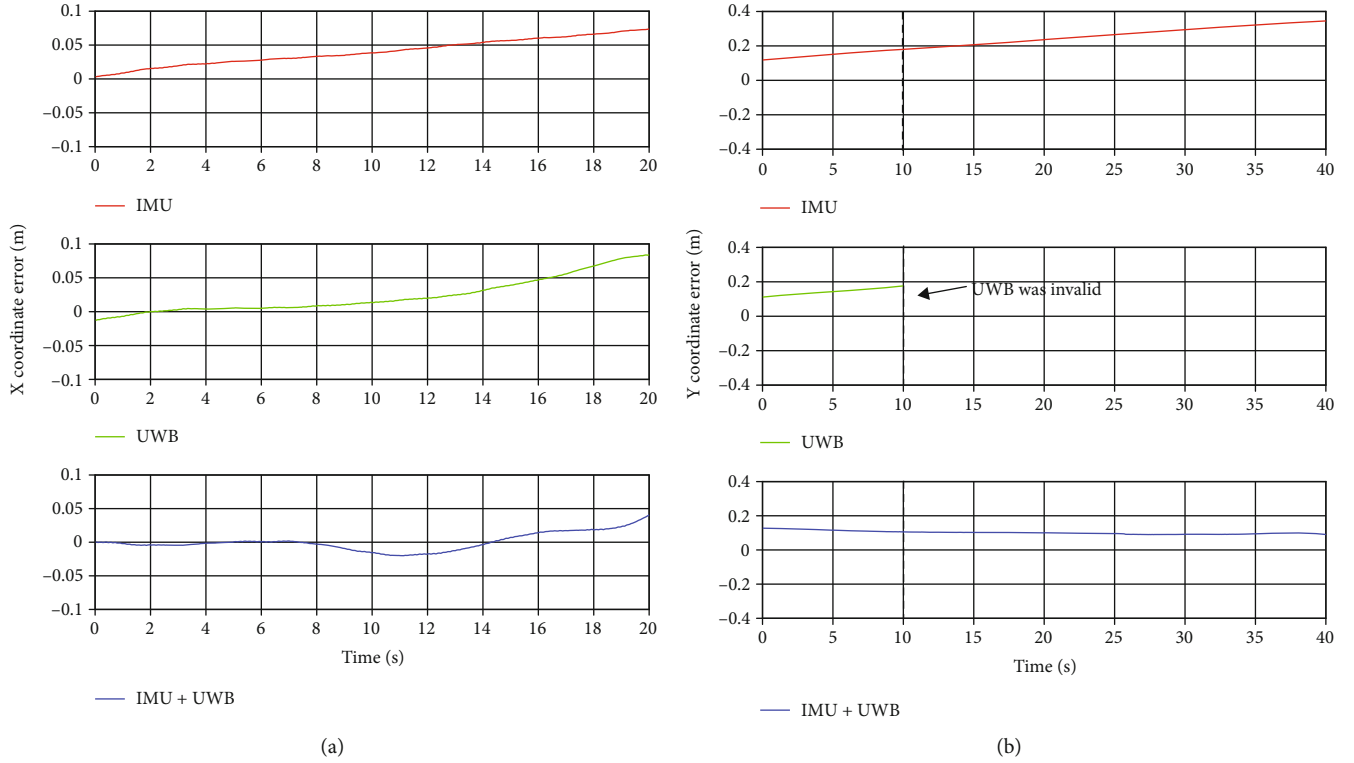
FIGURE 19: Robot coordinate errors by three positioning methods: (a) X coordinate errors; (b) Y coordinate errors.

In the process of multisensor data fusion and robot positioning, no matter which set of positioning data $((x_c, y_c)$ or $(\dot{x}_c, \dot{y}_c))$ is acquired, it will be fed to the Kalman filter to estimate the robot's position and velocities. At the same time, the covariance matrix of the Kalman filter is updated. If the UWB positioning signal is temporarily lost or interfered, the robot can still estimate its position using the data from IMU and encoders. Certainly, the robot can also complete the positioning operation only with UWB positioning data. Based on the method of data fusion and position estimation, multiple positioning data are mutually corrected and supplemented to improve robot positioning accuracy.

5. Experiments and Results

5.1. UWB Ranging Correction Experiments.
In the UWB ranging correction experiments, the UWB measured distance and the actual distance were collected and data fitting was performed to correct the ranging error. In the LOS environment, the distance between a UWB tag and a UWB anchor was measured by the DS-TWR method. In order to verify the reliability of the method, the experiment was implemented in two different environments: indoor correction experiment (Figure 13(a)) and outdoor correction experiment (Figure 13(b)). It is worth to note that there was no obstruction between the UWB tag and the UWB anchor.

Table 1 shows the comparison of multiple UWB measurement distances and actual distances. After data fitting, the correction equations for UWB indoor and outdoor ranging are

$$d_{t\_in} = 0.9846 \times d_m + 0.3327,$$
$$d_{t\_out} = 0.9898 \times d_m + 0.1564,$$

(36)

where d_m is the original UWB measured value, d_{t_in} and d_{t_out} are correctional value in indoor and outdoor environments, respectively. The correction equation is used to reduce the ranging error and improve the measurement accuracy.

Figure 14 shows the curves of the original UWB ranging error and the correctional UWB ranging error. The correctional ranging error is significantly smaller than the original ranging error. In the indoor environment (Figure 14(a)), the root mean square error (RMSE) of the original UWB ranging data is 0.19 m, and the RMSE of correctional UWB data is 0.04 m. In the outdoor environment (Figure 14(b)), the RMSE of the original and correctional UWB ranging data is 0.08 m and 0.01 m, respectively.

Compared with the outdoor environment, the indoor environment has lower ranging accuracy due to UWB multipath interference. The experimental results indicate that the correction method of data fitting can effectively reduce the UWB ranging error, especially in the case of severe UWB multipath interference. The maximum indoor ranging error is reduced to 0.1 m from original 0.4 m. Similarly, the maximum error of outdoor UWB ranging is corrected from 0.17 m to 0.04 m.
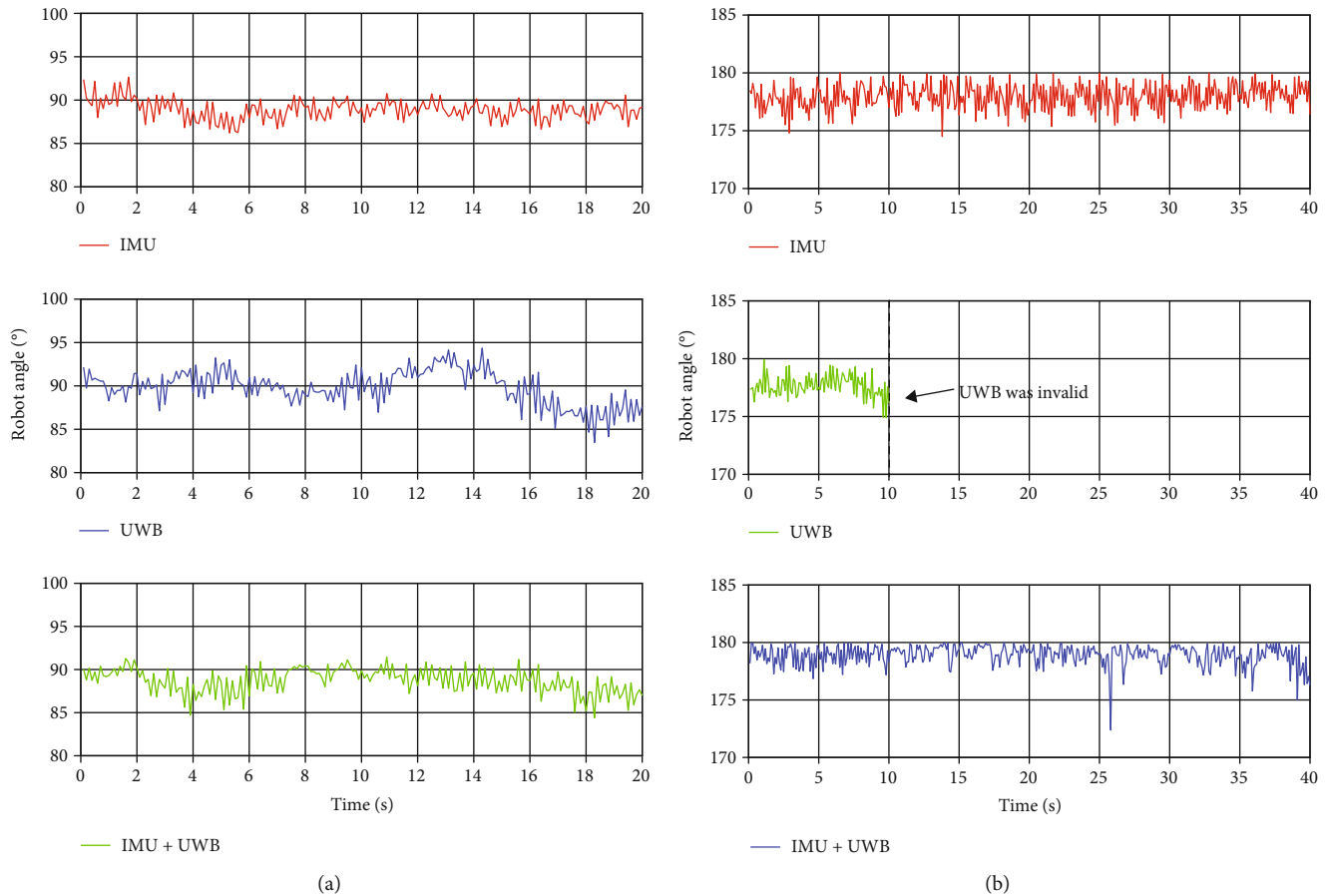
FIGURE 20: Changing curves of the robot's heading angle: (a) SM stage; (b) ME stage.

5.2. UWB Availability Experiment. In the indoor environment, the UWB robot tag may exceed the visible range of UWB anchors due to the occlusion of walls or obstacles, which will cause inaccurate UWB data at this case. As a result, during UWB positioning progress, the first path power of the UWB anchor signal will be captured to confirm the availability of UWB data. If the communication power of a UWB anchor is too low, the ranging data of this UWB anchor is unreliable; then, it will be discarded.

Figure 15 shows the first path power and ranging results of four UWB anchors as the robot's position changed. When the robot was in Position I, Anchor 0 was blocked and interfered; its signal power was very low. At this stage, the ranging data of Anchor 0 was unstable, so it was discarded. When the robot moved to Position II, Anchor 0 was in the visible range of the robot, and its signal power gradually increased. All UWB anchors had stable signals, and their positioning data were available. When the robot left Position II and entered Position III, Anchor 3 left the visible area of the robot. The signal power of Anchor 3 gradually decreased, and its ranging data was discarded due to inaccuracy.

In the process of robot positioning, if the first path power of a UWB anchor is lower than -95 dBm, the robot positioning system will discard the ranging data of this

anchor. When the total number of available UWB anchors is less than three, the UWB positioning status of the robot will be considered invalid at this time.

5.3. Multisensor Positioning Experiments. Multisensor positioning experiments were carried out to verify effects of the data fusion and robot positioning by UWB, IMU, and encoders. Figure 16 shows the experimental equipment and environment, the UWB robot tag was installed on the mobile robot, and three UWB anchors (Anchor 1, Anchor 2, and Anchor 3) were distributed around. Meanwhile, the robot's velocities were calculated based on the IMU sensor (MPU9250) and encoders. The laser tracker was used to record the running trajectory and position coordinates of the robot in real time. Within the coverage of UWB signals, multisensor data based on UWB, IMU, and encoders were simultaneously captured and used; outside the coverage of UWB signals, the robot could only use IMU and encoders for positioning.

The given running path of the mobile robot is shown in Figure 17. The robot started from point *S*, passed through point *M*, and finally reached point *E*. In the experimental environment, zone *A* was completely covered by UWB signals. UWB signals in Zone*B*were slightly disturbed by walls, and those in Zone *C* were invalid due to severe interference.

Three positioning methods were tested and compared in the experiments:

(1) Positioning only by IMU and encoders (IMU positioning);

(2) Positioning only by the UWB subsystem (UWB positioning);

(3) Positioning by IMU, encoders, and UWB subsystem (IMU+UWB positioning).

In the multisensor positioning experiments, the robot positioning system adopted these three different positioning methods to estimate the coordinates and angle of the mobile robot. The robot coordinate system XOY was established, with Anchor 1 as the coordinate origin. Figure 18 shows the coordinate system and the running trajectories of the robot by three positioning methods. The coordinates of three UWB anchors were (0, 0), (2.0, 3.0), and (1.0, 6.0). In the first stage, the robot started from point S (1.0,1.0) and reached point M (1.0, 5.6) along the direction of the Y coordinate axis. In the second stage, the robot rotated 90° and then moved along the direction of the X coordinate axis to the point E (-8.0, 5.6).

As shown in Figure 18, the red lines represent the running trajectory of the robot using IMU positioning. In the whole process, the robot coordinates can be estimated and used, but as time accumulates, the positioning error gradually increases. The blue lines are the running trajectory of the robot using UWB positioning. In the Zone C, the robot positioning fails because the UWB signals are invalid. The green lines represent the running trajectory of the robot using IMU+UWB positioning. In Zone A and Zone B, the robot uses data fusion of UWB, IMU, and encoders for positioning, and the robot can still complete positioning using IMU and encoder data in Zone C.

Figure 19 shows the coordinate error of the robot using three positioning methods. Figure 19(a) shows the X coordinate errors when the robot moved from point S to point M (SM stage). In the initial stage, the error of IMU positioning is very small, but it keeps increasing over time. The error of UWB positioning is the largest. Since the UWB data has been used to correct the heading angle of IMU, the error of IMU+UWB positioning has some fluctuations while its high accuracy is guaranteed. The maximum X coordinate error of IMU+UWB positioning is less than 0.05 m.

Figure 19(b) shows the Y coordinate error when the robot moved from point M to point e (ME stage). The error of IMU positioning keeps increasing, and the maximum error is close to 0.4 m. As UWB signals in Zone B are disturbed, the error of UWB positioning continues to increase. In Zone C, the method of UWB positioning fails and cannot estimate the robot coordinates. In the whole process, the error of IMU+UWB positioning is relatively small. Even if the UWB signals are invalid in Zone C, the robot can still complete the positioning with high accuracy. The Y coordinate error is kept within 0.16 m.

Based on mathematical statistics, the RMSEs of IMU positioning, UWB positioning and IMU+UWB Positioning are 0.1972 m, 0.0902 m, and 0.0821 m, respectively. The experimental results indicate that the multisensor fusion positioning method based on UWB, IMU, and encoders has higher positioning accuracy and stability. The multisensor positioning method uses different sensor data to correct and supplement each other, thus having stronger anti-interference ability. Even if one sensor fails, the robot positioning can still operate normally based on other sensors.

Figure 20 shows changing curves of the robot's heading angle in the experiments. In the SM stage (Figure 20(a)), the desired heading angle of the robot is 90°. In a short time, the heading angle of IMU positioning is more stable, but the angle error cannot be corrected automatically. The heading angle of UWB Positioning is calculated by coordinates with some large fluctuations. This is because UWB data is used to correct IMU errors, the heading angle of IMU+UWB positioning also has some slight fluctuations. In the ME stage (Figure 20(b)), the desired heading angle of the robot is 180°. The cumulative angle error of the IMU positioning gradually increases, which will cause greater positioning error. As UWB signals are interfered, UWB positioning has more angle errors, and it fails in Zone C, whereas the heading angle of IMU+UWB positioning is more stable. It can still operate normally in Zone C.

The experimental results indicate that the positioning accuracy of the multisensor positioning method has been greatly improved. The UWB data can correct the cumulative angle error of IMU, and the heading angle is more stable. When UWB signals are interfered or invalid, the robot can still use IMU and encoder data for positioning, which has stronger environmental adaptability.

6. Discussion of the Experiments

Compared with the single-sensor positioning method, the positioning system based on UWB and multisensors can improve the positioning accuracy and adaptability of the robot. Through data fitting, the robot positioning system can effectively reduce the UWB ranging error caused by multipath interference. The indoor and outdoor maximum ranging errors are increased to 0.1 m and 0.04 m, respectively. According to the first path power of the UWB anchor signals, the robot positioning system can judge the availability of UWB positioning data and prevent the error positioning data from being used. Meanwhile, the positioning system can select stable UWB anchors to collect positioning data.

In general, the developed robot positioning system based on data fusion and error correction has the following advantages:

(1) The mutual correction of multisensor data improves the stability and accuracy of positioning data. During the movement of the robot, IMU heading angle can be corrected by UWB data, which reduces the cumulative angle error

(2) Based on multisensor data fusion, the robot positioning system can achieve higher-precision positioning. When multisensor data are available

simultaneously, the positioning error of the robot is less than 0.05 m

(3) The robot positioning system has better anti-interference ability and environmental adaptability. Even if UWB positioning data is disturbed, the robot can still use IMU and encoder data for positioning, and the maximum positioning error is within 0.16 m

7. Conclusion

This paper presents a position estimation and positioning error correction method of mobile robot based on UWB, IMU, and encoders. The robot positioning system includes a mecanum wheeled mobile robot, a remote control computer, and UWB positioning subsystem. UWB ranging error is corrected by the DS-TWR method and data fitting. The UWB positioning system is composed of UWB anchors, and a UWB robot tag can calculate robot position coordinates. The multisensor positioning method can improve the robot positioning accuracy. The robot coordinates (x_c, y_c) are captured by the UWB subsystem, and the robot velocities ($\dot{x}_c$, $\dot{y}_c$) are calculated by the IMU and encoders. Multisensor data are continuously fed to the Kalman filter to estimate the robot position. In the positioning process, UWB data are also used to reduce the cumulative angle error generated by the IMU. In the UWB ranging correction experiments, the correction equation after data fitting can effectively improve the UWB ranging accuracy. The first path power of the UWB anchor signal is captured to confirm the availability of UWB data. In the multisensor positioning experiments, the multisensor positioning method (IMU + WB positioning) has higher accuracy and stability than a single-sensor positioning method (IMU positioning and UWB positioning). The experimental results indicate that the multisensor positioning method based on UWB, IMU, and encoders can effectively improve the positioning accuracy of the mobile robot, and this method has stronger anti-interference ability.

In the future, we will optimize the robot positioning system to improve the positioning accuracy. For instance, more different types of sensors can be combined in the process of robot positioning and error correction. Besides, more multisensor positioning tests will be carried out to expand its application scope.

Data Availability

All data included in this study are available upon request by contact with the corresponding author.

Conflicts of Interest

The authors declare no conflict of interest.

Acknowledgments

References

[1] S. S. Saab and Z. S. Nakad, "A standalone RFID indoor positioning system using passive tags," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 5, pp. 1961–1970, 2011.

[2] M. Z. A. Bhuiyan, G. Wang, and A. V. Vasilakos, "Local area prediction-based mobile target tracking in wireless sensor networks," *IEEE Transactions on Computers*, vol. 64, no. 7, pp. 1968–1982, 2015.

[3] A. Oka and L. Lampe, "Distributed target tracking using signal strength measurements by a wireless sensor network," *IEEE Journal on Selected Areas in Communications*, vol. 28, no. 7, pp. 1006–1015, 2010.

[4] A. Alarifi, A. Al-Salman, M. Alsaleh et al., "Ultra wideband indoor positioning technologies: analysis and recent advances," *Sensors (Switzerland).*, vol. 16, no. 5, p. 707, 2016.

[5] S. Park and S. Hashimoto, "Autonomous mobile robot navigation using passive RFID in indoor environment," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 7, pp. 2366–2373, 2009.

[6] L. Zhang, Z. Chen, W. Cui et al., "WiFi-based indoor robot positioning using deep fuzzy forests," *IEEE Internet of Things Journal*, vol. 7, no. 11, pp. 10773–10781, 2020.

[7] S. J. Kim and B. K. Kim, "Accurate hybrid global self-localization algorithm for indoor mobile robots with two-dimensional isotropic ultrasonic receivers," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 10, pp. 3391–3404, 2011.

[8] Q. Fan, B. Sun, Y. Sun, Y. Wu, and X. Zhuang, "Data fusion for indoor mobile robot positioning based on tightly coupled INS/UWB," *Journal of Navigation*, vol. 70, no. 5, pp. 1079–1097, 2017.

[9] J. González, J. L. Blanco, C. Galindo et al., "Mobile robot localization based on ultra-wide-band ranging: a particle filter approach," *Robotics and Autonomous Systems*, vol. 57, no. 5, pp. 496–507, 2009.

[10] Y. Xu, Y. S. Shmaliy, C. K. Ahn, G. Tian, and X. Chen, "Robust and accurate UWB-based indoor robot localisation using integrated EKF/EFIR filtering," *IET Radar, Sonar and Navigation*, vol. 12, no. 7, pp. 750–756, 2018.

[11] K. C. Cheok, M. Radovnikovich, P. K. Vempaty, G. R. Hudas, J. L. Overholt, and P. Fleck, "UWB tracking of mobile robots," in *IEEE Int. Symp. Pers. Indoor Mob. Radio Commun*, pp. 2615–2620, PIMRC, 2010.

[12] C. I. Transceivers, B. G. Yu, G. Lee et al., "A time-based angle-of-arrival sensor using CMOS IR-UWB transceivers," *IEEE Sensors Journal*, vol. 16, no. 14, pp. 5563–5571, 2016.

[13] T. Gigl, G. J. M. Janssen, V. Dizdarević, K. Witrisal, and Z. Irahhauten, "Analysis of a UWB indoor positioning system based on received signal strength," in *4th Work. Positioning, Navig. Commun. 2007*, pp. 97–101, WPNC'07- Work. Proc. 2007, 2007.

[14] J. Y. Lee and S. Yoo, "Large error performance of UWB ranging in multipath and multiuser environments," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 4, pp. 1887–1895, 2006.

[15] K. C. Ho, "Bias reduction for an explicit solution of source localization using TDOA," *IEEE Transactions on Signal Processing*, vol. 60, no. 5, pp. 2101–2114, 2012.

[16] B. Hanssens, D. Plets, E. Tanghe et al., "An indoor localization technique based on ultra-wideband AoD/AoA/ToA estimation," in *2016 IEEE Antennas Propag. Soc. Int. Symp*, pp. 1445-1446, APSURSI 2016- Proc, 2016.

[17] C. WannDer, Y. J. Yeh, and C. S. Hsueh, "Hybrid TDOA/AOA indoor positioning and tracking using extended Kalman filters," *IEEE Veh. Technol. Conf.*, vol. 3, pp. 1058–1062, 2006.

[18] E. Lagunas, M. Najar, and M. Navarro, "UWB joint TOA and DOA estimation," in *Proc. -2009 IEEE Int. Conf. Ultra-Wideband*, pp. 839–843, ICUWB 2009, 2009.

[19] A. R. Jiménez Ruiz, F. Seco Granja, J. C. Prieto Honorato, and J. I. Guevara Rosas, "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 1, pp. 178–189, 2012.

[20] X. Yan, H. Guo, M. Yu, Y. Xu, L. Cheng, and P. Jiang, "Light detection and ranging/inertial measurement unit-integrated navigation positioning for indoor mobile robots," *International Journal of Advanced Robotic Systems*, vol. 17, no. 2, p. 172988142091994, 2020.

[21] Y. Xu, T. Shen, X. Y. Chen, L. L. Bu, and N. Feng, "Predictive adaptive Kalman filter and its application to INS/UWB-integrated human localization with missing UWB-based measurements," *International Journal of Automation and Computing*, vol. 16, no. 5, pp. 604–613, 2019.

[22] L. Zwirello, C. Ascher, G. F. Trommer, and T. Zwick, *Study on UWB / INS Integration Techniques*, pp. 13–17, 2011.

[23] Z. Liu, J. Li, A. Wang, X. Cheng, and A. Wang, "Design and implementation of UWB/MIMU tightly-coupled system for indoor positioning," in *Proc. 5th IEEE Conf. Ubiquitous Positioning, Indoor Navig. Locat. Serv*, pp. 1–7, UPINLBS 2018, 2018.

[24] D. Neirynck, E. Luk, and M. McLaughlin, "An alternative double-sided two-way ranging method," in *Proc. 2016 13th Work. Positioning, Navig. Commun*, WPNC 2016, 2017.

[25] G. Shen, R. Zetik, and R. S. Thomä, "Performance comparison of TOA and TDOA based location estimation algorithms in LOS environment," in *5th Work. Positioning, Navig. Commun. 2008*, pp. 71–78, WPNC'08. 2008, 2008.

[26] A. Jouybari, H. Amiri, A. A. Ardalan, and N. K. Zahraee, "Methods comparison for attitude determination of a lightweight buoy by raw data of IMU," *Meas. J. Int. Meas. Confed.*, vol. 135, pp. 348–354, 2019.

Research Article

Simulation Analysis of Eddy Current Testing Parameters for Surface and Subsurface Defect Detection of Aviation Aluminum Alloy Plate

Weiquan Deng [1], **Jun Bao** [2], **Siqi Luo** [3], and **Xin Xiong** [3]

¹*Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China*
²*Faculty of Civil Aviation and Aeronautics, Kunming University of Science and Technology, Kunming 650500, China*
³*Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China*

Correspondence should be addressed to Xin Xiong; xinxiong@kust.edu.cn

In the process of eddy current testing (ECT) of surface and subsurface defects of aviation aluminum alloy plates, the setting of parameters is important to the test results. Inappropriate test parameters can cause false detection or even missed detection of defects. To address this problem, the effects of probe type, coil size, and excitation frequency on the accurate identification and quantitative evaluation of surface and subsurface defect detection were studied and analyzed in this study to determine the best testing parameters. The experimental results show that the absolute probe with an outer radius of 3.3 mm has better detection performance for aviation aluminum alloy plate defects. There are different optimal excitation frequency ranges for the surface and subsurface defects. An excitation frequency of 80 kHz to 90 kHz can be used for the detection of unknown defects.

1. Introduction

Aluminum alloy materials are widely used in the manufacture of civil aircraft flaps, skins, and other structural parts owing to their advantages, such as low density, high strength, good processability, and strong corrosion resistance. According to the statistics, in ordinary civil aviation aircraft, the use of aluminum alloy materials exceeds 60% of the total weight of the aircraft. However, owing to the influence of the service environment and long-term high-load operation, various corrosion and fatigue crack defects inevitably occur in aircraft. The existence of defects in aircraft can cause safety hazards to the aircraft structure as well as major safety threats and economic losses. Therefore, it is critical to detect, identify, and accurately quantify the defects in aluminum alloy plates over time.

Among various nondestructive testing (NDT) technologies, eddy current testing (ECT) is the most suitable for the detection of corrosion and fatigue cracks in aircraft aluminum alloy plates owing to its fast detection speed, wide

detection range, easy automation, and higher detection performance for surface and subsurface defects of detected objects [1–5]. By combining principal component analysis and k-means classification, Kim et al. conducted feature extraction of a deeper crack in an aircraft to realize defect detection and location [6]. Based on the time-frequency analysis of the pulsed eddy current defect detection signal, combined with k-means clustering and expectation maximization, Hosseini and Lakis realized the automatic detection of the distribution of subsurface defects in each layer of a multilayer aluminum alloy plate structure [7]. He et al. realized the automatic detection of layered defects of an aircraft two-layer aluminum alloy plate structure based on the constructed defect feature and support vector machine classification algorithm and further studied the influence of different lift-off distances on the detection results [8]. Li et al. studied the subsurface defect detection of aluminum alloy plates based on pulse-modulated eddy currents, which improved the detection performance of subsurface defects [9]. For the detection of subsurface defects in aluminum

alloy plates, Perumal et al. conducted a simulation analysis from multiple excitation frequencies, and combined with the optimization of excitation frequency, the depth of different subsurface defects can be located [10]. Yan et al. used the improved Canny algorithm to identify the edge of the defect ECT image of aluminum alloy materials, which improved the accuracy of corrosion edge detection under coating [11]. Considering the influence of edge effects on defect detection, Xie et al. conducted finite element simulation from the design of probe parameters to improve the ability of edge defect detection [12]. In addition, for defect detection, some studies have been conducted on the combination of the probe type and defect scanning method.

However, in the existing research on defect detection of aluminum alloy plate materials, most of the studies are based on single detection parameter optimization and detection signal processing algorithms for defect detection and quantitative analysis [13–16], thus insufficient accuracy of subsurface defect detection and defect edge extraction. Starting from the detection parameters, in this study, we analyze the effects of probe type, coil size, and excitation frequency on the surface and subsurface defect detection of aviation aluminum alloy plates and construct a finite element model to optimize the detection parameters. The rest of this paper is organized as follows: First, the theoretical method of probe coil design is analyzed in Section 2. Second, the finite element model of the ECT is constructed, and several testing parameters are set in Section 3. Next, the simulation results are analyzed in detail in Section 4. Finally, conclusions and further research are outlined in Section 5.

2. Theoretical Method

ECT is an NDT method based on the change in the electromagnetic properties of the tested conductor to analyze its properties and defects. This method is based on electromagnetism theory. When a current of a certain frequency flows into the excitation coil, under the action of electromagnetic induction, the induced eddy current is generated in the detected conductor. Furthermore, the induced eddy current generates a magnetic field and reacts on the test coil, thus affecting the voltage and current of the coil.

The ECT coil is composed of multiple turns of metal enameled wire; thus, the coil itself has inductance. In addition, there is resistance between the winding enameled wire, as well as a coupling capacitor between each turn of the coil. In actual detection and calculation, the capacitance distributed between each turn of the coil is usually ignored. The coil can be expressed as a series circuit of inductance L_1 and resistance R_1, and the complex impedance of the coil itself can be expressed as

$$Z_1 = R_1 + j\omega L_1. \tag{1}$$

Under the action of the induced eddy current excited by the excitation coil, the conductor can be expressed as a secondary circuit composed of inductance and resistance in series. The complex impedance of the coil is affected by the equivalent impedance Z_z of the conductor.

$$Z_z = R_z + j\omega L_z = \frac{\omega^2 M^2}{R_2{}^2 + \omega^2 L_2{}^2} R_2 + j\omega \frac{\omega^2 M^2}{R_2{}^2 + \omega^2 L_2{}^2} L_2. \tag{2}$$

L_2 and R_2 are the equivalent inductance and resistance of the conductor under test, M is the mutual inductance between the coil and the conductor, and M can be expressed as

$$M = k(l)\sqrt{L_1 L_2}. \tag{3}$$

$k(l)$ is the coupling coefficient, which is related to the lift-off distance between the coil and conductor. The change in the electromagnetic properties of the conductor affects its equivalent impedance Z_z and, consequently, affects the complex impedance of the detection coil. The conductivity, permeability, defect, and thickness (sheet) of the conductor can be obtained by collecting and analyzing the impedance signal of the detection coil.

The design of the ECT probe coil should follow the appropriate inductance value, resistance value, and higher Q value (i.e., quality factor, $Q = \omega L/R$). For a multiturn coil, without considering the influence of the capacitance between the turns of the enameled wire, the inductance L is proportional to the square of the number of turns N of the coil, that is,

$$L \propto N^2 = \frac{(r_2 - r_1)^2 h^2}{d^4}. \tag{4}$$

r_2, r_1, and h are the outer radius, inner radius, and height of the coil, respectively, and d is the diameter of the enameled wire.

The resistance of the coil itself can be calculated according to the following formula:

$$R = \rho \frac{l}{S} = 4\rho \frac{\pi(r_2{}^2 - r_1{}^2)h}{d^4}. \tag{5}$$

Combining Equations (3) and (5), the quality factor Q of the coil can be expressed as follows:

$$Q = \frac{\omega L}{R} \propto \frac{\omega h}{\rho} \cdot \frac{r_2 - r_1}{r_2 + r_1}. \tag{6}$$

It can be observed from the above equation that when the wire of the coil is determined, Q is determined by the excitation frequency and coil size. In the defect detection process, if the coil is significantly small, the excitation current of the coil must be reduced accordingly. Moreover, if the excitation current is significantly small, the detected induction signal reduces, which further causes the defect signal to be submerged in noise, thus the inability to obtain an effective detection signal. However, if the coil radius is increased, a coil that is considerably large creates a difference between the detected defect edge and the actual defect edge; thus, defect edge detection becomes difficult.

For the ECT of surface and subsurface defects of aviation aluminum alloy plates, in this study, we simulated and
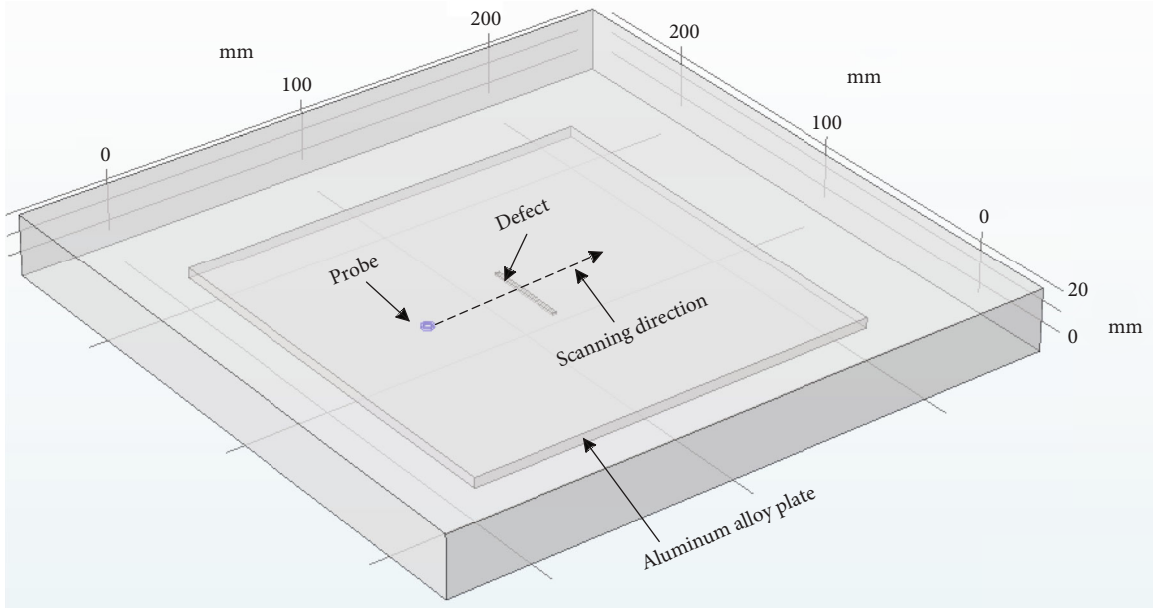
FIGURE 1: Finite element simulation model of ECT.

TABLE 1: Electromagnetic properties of the simulation model.

Material	Electromagnetic properties	
	Conductivity (MS/m)	Relative permeability
6000 series aluminum alloys specimen	26	1
Probe coil made of copper	58	0.99
Vacuum computing region	0	1

optimized the probe type, coil radius, and excitation frequency to improve the accuracy of defect detection and defect edge recognition.

3. Simulation Experiment Modeling

To accurately analyze the effects of probe type, coil size, and excitation frequency on the accuracy of surface and subsurface defect detection and edge recognition of aviation aluminum alloy plates, a finite element model of ECT based on COMSOL multiphysical field simulation platform was constructed in this study, as shown in Figure 1. The simulation model included an aluminum alloy specimen, eddy current probe, and vacuum domain. In this model, different parameters, such as the probe type, coil size, and excitation frequency, were simulated and calculated. The depth and position of the defects in the specimen and the size of the coil were set according to different research contents.

The specimen chosen for this study was a 6000 series aluminum alloy plate with a thickness of 5 mm that contains defects of different sizes. The coil was composed of copper enameled wire and placed above the specimen. The distance between the coil and the specimen was set to 1 mm. The electromagnetic properties of simulation model materials are shown in Table 1.

Before the simulation calculation, the model should be meshed and divided into multiple elements. It is necessary to consider the accuracy and efficiency of calculation simultaneously when meshing. The smaller the element is, the higher the calculation accuracy will be, and the calculation time will increase accordingly. Considering that the inspection object constructed was an aluminum alloy thin plate, a smaller mesh division did not significantly increase the calculation time; thus, extremely fine mesh generation was adopted in this study, as shown in Figure 2.

Based on meshing, the combination of parametric scanning and frequency domain analysis was used for simulation calculation. The scanning path of defects in the simulation process is shown in Figure 1. Assuming the defect as the center, the left and right sides were both 25 mm for scanning of the defect, and the scanning step is 0.5 mm. After the simulation solution, the coil impedance at different positions and frequencies was obtained.

3.1. Simulation of Coil Structure. In the process of fatigue crack detection and analysis of aircraft aluminum alloy plates, different probe types have an impact on the accuracy of defect detection and edge recognition. The common eddy current probe selection includes the absolute and differential types. As shown in Figure 3, the excitation and receiving coils are the same as those in the absolute probe. In Figure 4, the excitation of the differential probe consists of two inversely connected coils, and the receiving coil is placed between the two excitation coils (differential excitation).
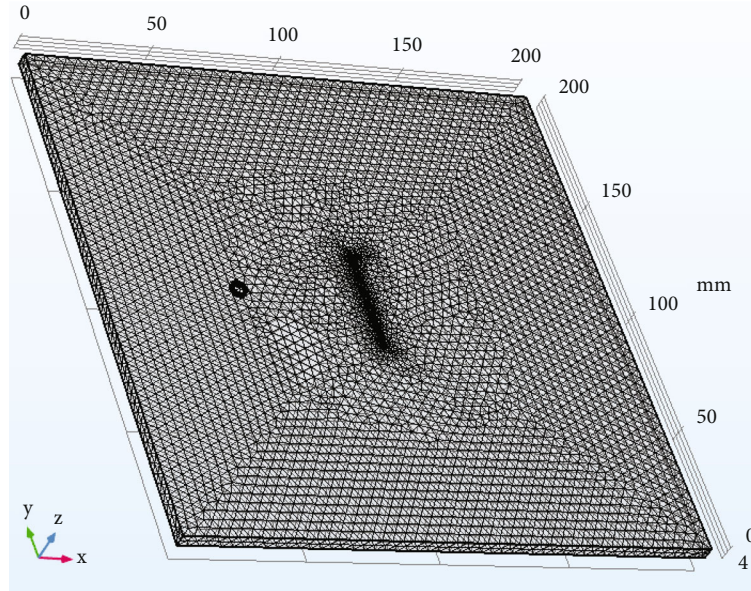
FIGURE 2: Meshing of the simulation model.

Another form of differential probe is that the receiving coil is completed by two reverse-connected coils, and the excitation coil is placed in the middle of the receiving coil, that is, differential reception. Compared with the differential receiving probe, the differential excitation probe exhibited stronger anti-interference. Therefore, the differential excitation probe was used for the simulation experiment analysis of defect detection in this study.

According to Equation (6), it can be observed that the quality factor is determined by the inductance and resistance of the coil, which can be further transformed into the relationship with the coil size. Considering that the detection object was the defect of the aluminum alloy plate, the change in the coil height did not affect the detection of the defect edge. The quality factor was further simplified as the relationship between the inner and outer diameters of the coil. The inner and outer radius ranges were 1–4 mm and 2–5 mm, respectively, and the corresponding coil quality factor was calculated. The results are shown in Figure 5.

From the simplified calculation results, the quality factor was affected by the outer radius. The larger the outer radius, the higher the quality factor. The quality factor was not affected by the change of coil inner radius. When the ratio of inner radius to outer radius was less than 0.6, it can be considered to be in an appropriate range. However, in the process of defect detection, we should be able to accurately identify the existence of defects, as well as accurately identify the edge of defects from the detection signal, in order to make quantitative analysis of defects. Therefore, the larger the outer diameter of the coil used for defect detection is not the better.

In the defect scanning shown in Figure 1, it is necessary to accurately identify the defect and its edge information from the detection signal. To study the influence of different probe selections on defect edge recognition, 2 mm and 20 mm defects with different widths were set, and the simulation parameters are listed in Table 2.
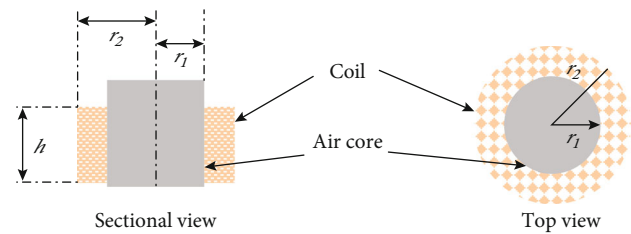
FIGURE 3: Schematic diagram of absolute probe.

To study the coil size of probe suitable for crack defect detection of aviation aluminum alloy plate, the scanning experiments of coils with different sizes were carried out on the simulation model constructed in Figure 1. The experimental parameters are shown in Table 3; seven coil outer radii of different sizes are set. When the inner radius of the coil is 0.3~0.6 times of the outer radius, the appropriate quality factor Q of the coil can be guaranteed [17].

3.2. Simulation of Excitation Frequency. Owing to the skin effect of the eddy current, the penetration depth of the eddy current decreases sharply with an increase in the excitation frequency. If the excitation frequency is significantly small, the resolution of the ECT signal to defects becomes insufficient, making it difficult to accurately identify defects. If the excitation frequency is considerably large, deeper defects cannot be detected, and an excessive excitation frequency causes the eddy current signal to be more susceptible to noise interference. Therefore, it is necessary to determine a more suitable excitation frequency for the defect detection of aviation aluminum alloy plates.

Four surface defects and subsurface defects with different depths were constructed in an aviation aluminum alloy plate, and the defects were detected by frequency sweeping.
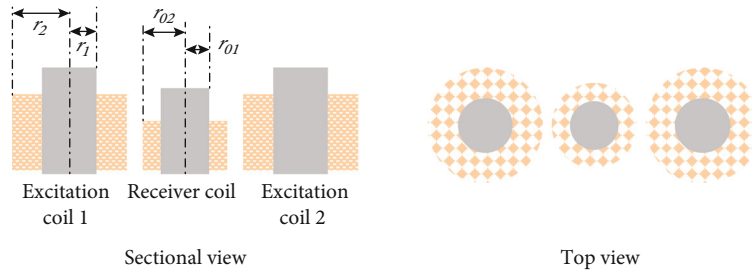
FIGURE 4: Schematic diagram of differential probe.

Inner radius r_1 (mm)
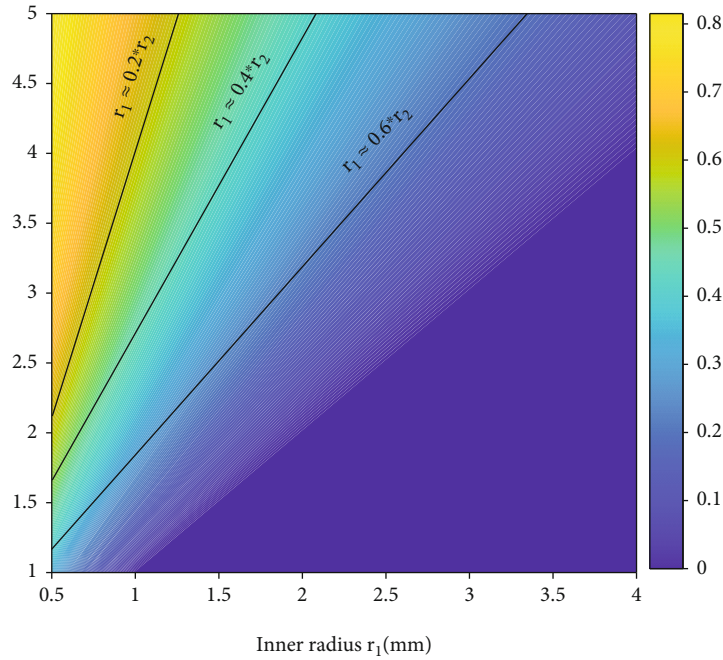
FIGURE 5: The relationship between coil quality factor and inner and outer radius.

TABLE 2: Parameters of absolute and differential probes for scanning defects.

Excitation frequency (kHz)	Excitation current (mA)	Coil parameters				Defect information (length × width × depth (mm))	
		Turns	r_1, r_{01} (mm)	r_2, r_{02} (mm)	h (mm)	#1	#2
100	50	300	2	3	1	40 × 2 × 1	40 × 20 × 1

The defects with different sizes are shown in Figure 6, and the simulation parameters are shown in Table 4.

4. Analysis of Experimental Results

4.1. Simulation Analysis of Coil Structure. According to the simulation parameters set in Table 2, the scanning results of different probe types for defect detection based on the constructed simulation model are shown in Figures 7 and 8, respectively. The horizontal and vertical axes represent the position and impedance value of the probe, respectively, and 0 in the horizontal axis represents the defect center. From the scanning results of the absolute probe on the defect in Figure 7, the signal in the entire defect area can be clearly iden-

tified, and the defect edge information can be accurately reflected in the scanned signal. However, in the scanning results of the differential probe for defects shown in Figure 8, the eddy current signal has multiple upward and downward trends in both defect areas, and the defect area and the edge information of the defect cannot be clearly identified from the scanned signal. It can be observed that the absolute probe is suitable for defect detection and quantitative analysis.

According to the above analysis, the detection signal is required to meet easy defect identification and edge detection simultaneously during defect detection, that is, accurate defect classification and quantitative analysis. For the defect detection signal obtained by simulation, the impedance difference Z_{diff} between the detection coil at the defect and

TABLE 3: Influence of different outer radius of coil on defect detection.

r_2 (mm)	r_1 (mm)	Excitation frequency (kHz)	Excitation current (mA)	Defect information (length × width × depth (mm))
2, 2.5, 3, 3.5, 4, 4.5, 5	$0.3 \sim 0.6 \times r_2$	100	50	$40 \times 1 \times 1$

(a) Surface defects

(b) Subsurface defects
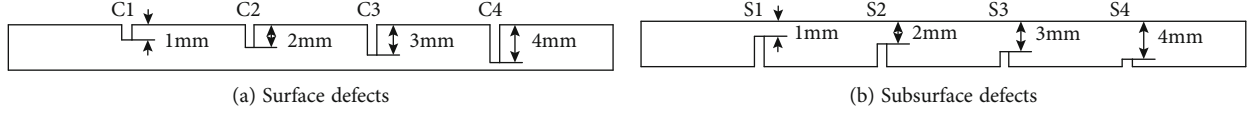
FIGURE 6: Schematic diagram of specimen defects.

TABLE 4: Influence of excitation frequency on defect detection.

Frequency range (kHz)	Step (kHz)	Depth of defect (mm)		r_2 (mm)	r_1 (mm)
		Surface defects	Subsurface defects		
10~1000	10	1, 2, 3, and 4	1, 2, 3, and 4	3	1.5
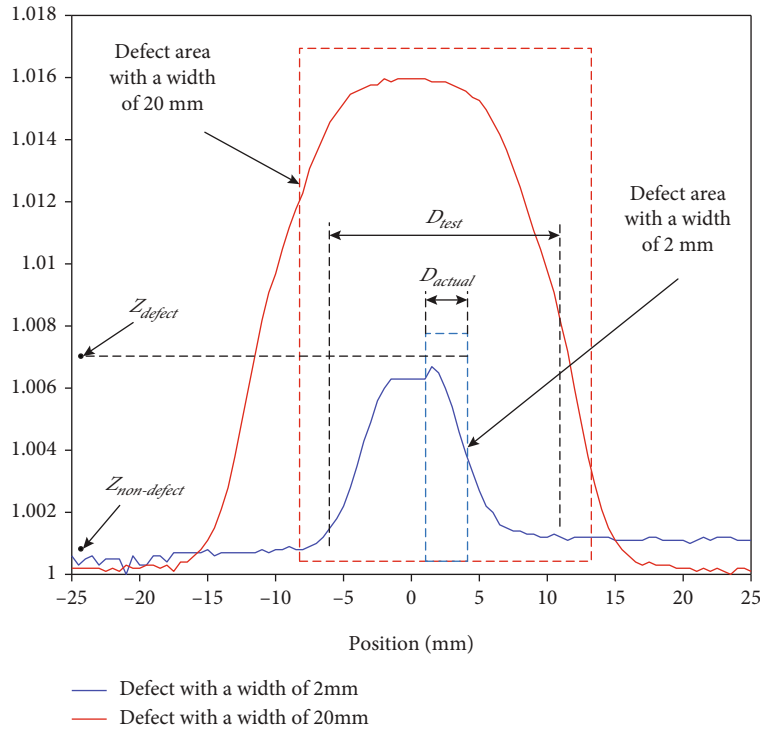
FIGURE 7: Scanning signal of absolute probe to defect.

nondefect and the proportion P_{ratio} of the scanning points of the actual defect area to the defect area determined by the scanning signal are calculated. According to the notes in Figure 7, Z_{diff} and R_{ratio} are calculated as follows:

$$Z_{\text{diff}} = Z_{\text{defect}} - Z_{\text{nondefect}},$$
$$P_{\text{ratio}} = \frac{D_{\text{actual}}}{D_{\text{test}}} \times 100. \tag{7}$$

The simulation solution was conducted according to the parameters set in Table 3, and Z_{diff} and P_{ratio} were calculated for the solution results of different outer radii. The simulation and calculation results are shown in Figure 9.

As shown in Figure 9, with the increase of the outer radius of the coil, the impedance difference Z_{diff} also gradually increases; that is, a larger coil radius is conducive to the detection of defects. However, the proportion P_{ratio} decreases with the increase of the coil radius; that is, the larger the coil
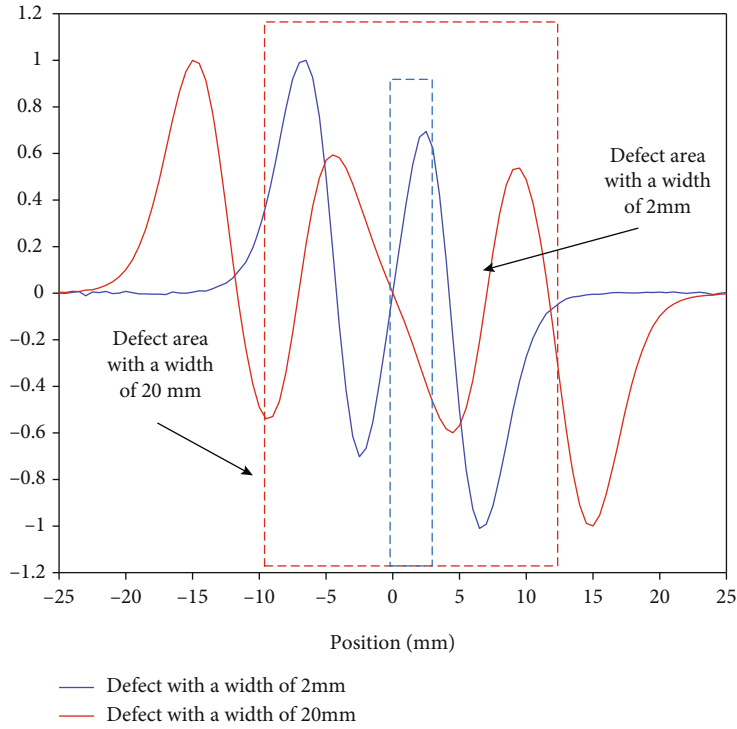
FIGURE 8: Scanning signal of differential probe to defect.
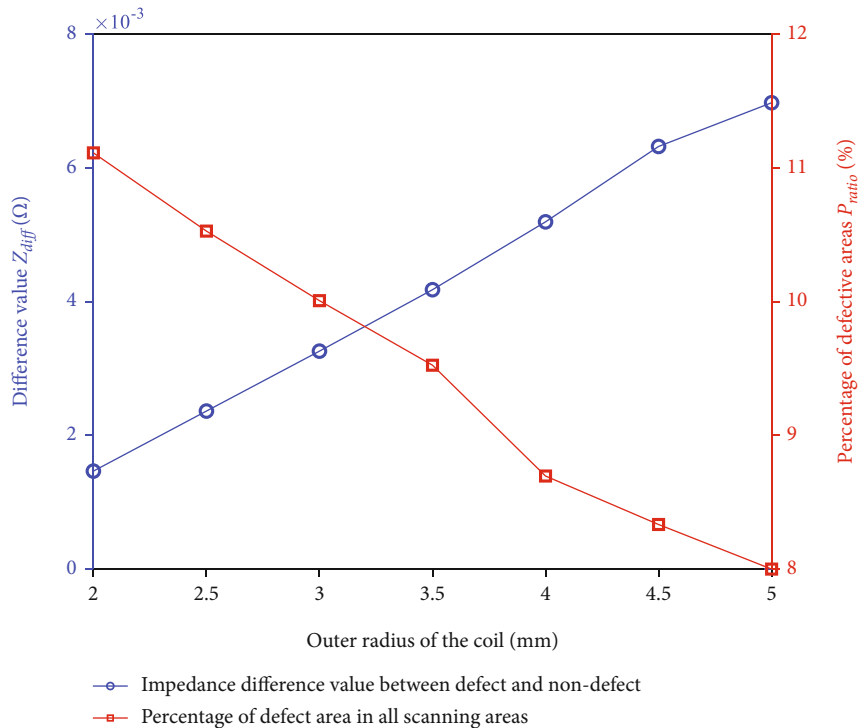
FIGURE 9: Simulation of coils with different sizes.

radius, the more difficult it is to identify the edge information of the defect from the detection signal. From the comprehensive analysis of the simulation results, when the outer radius of the coil is 3.3 mm, both defect detection and edge recognition can achieve higher accuracy.

4.2. Simulation Analysis of Excitation Frequency. According to the simulation parameters set in Table 4, the parametric scanning detection is conducted for surface and subsurface defects, as well as the impedance difference Z_{diff} between defect and nondefect is calculated using the constructed
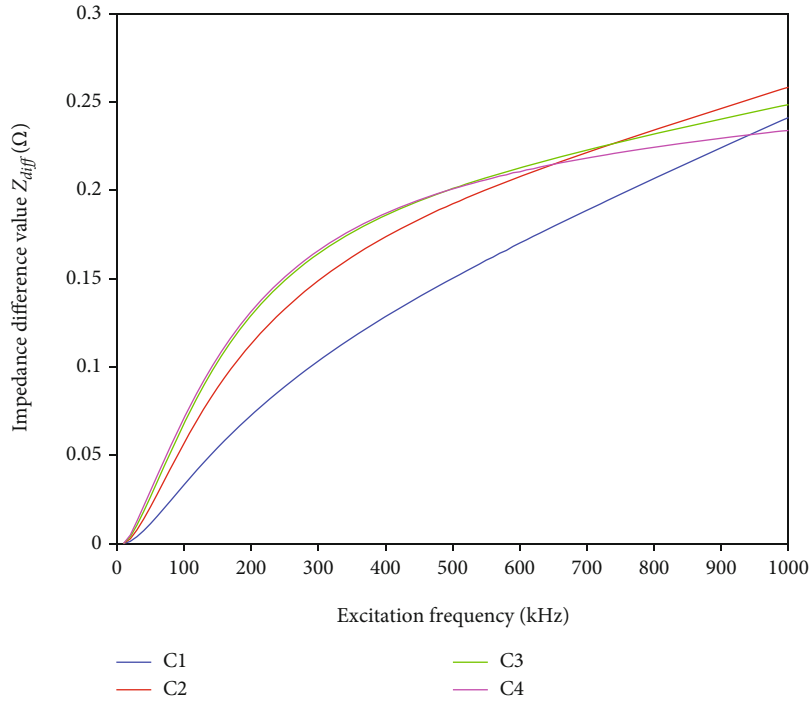
Figure 10: Influence of excitation frequency on surface defect detection.
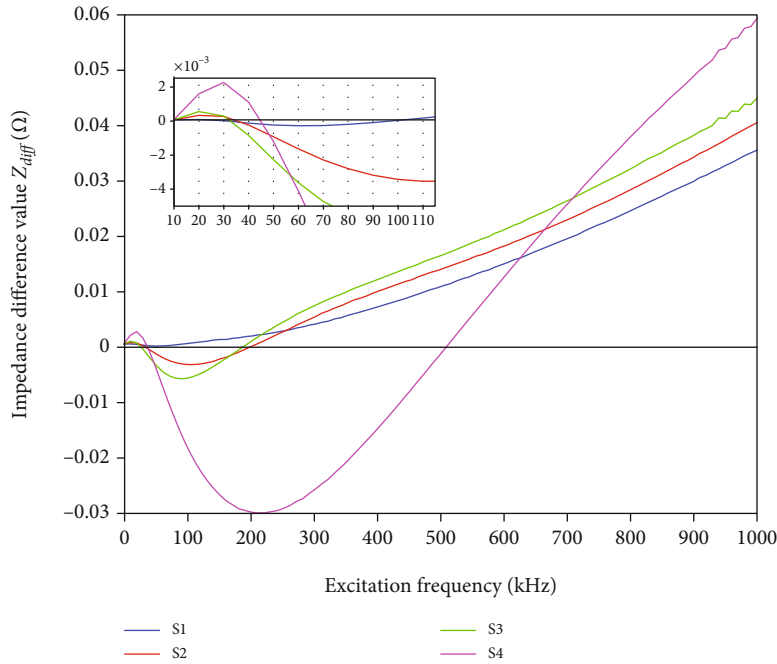
Figure 11: Influence of excitation frequency on subsurface defect detection.

simulation model. The detection results of surface defects and subsurface defects are shown in Figures 10 and 11, respectively.

In the surface defect simulation results shown in Figure 10, for defects with different depths, the impedance difference Z_{diff} increases with an increase in the excitation frequency. However, with the increase in surface defect depth, the impedance difference Z_{diff} tends to be stable at higher frequencies,

and there is no increasing trend. The deeper the defect, the earlier it shows a stable trend.

In the simulation results of subsurface defects shown in Figure 11, the change in impedance difference Z_{diff} is complex. Within a certain excitation frequency range, the impedance value at the defect is greater than that at the nondefect; that is, the impedance difference Z_{diff} is positive. With an increase in the excitation frequency, the
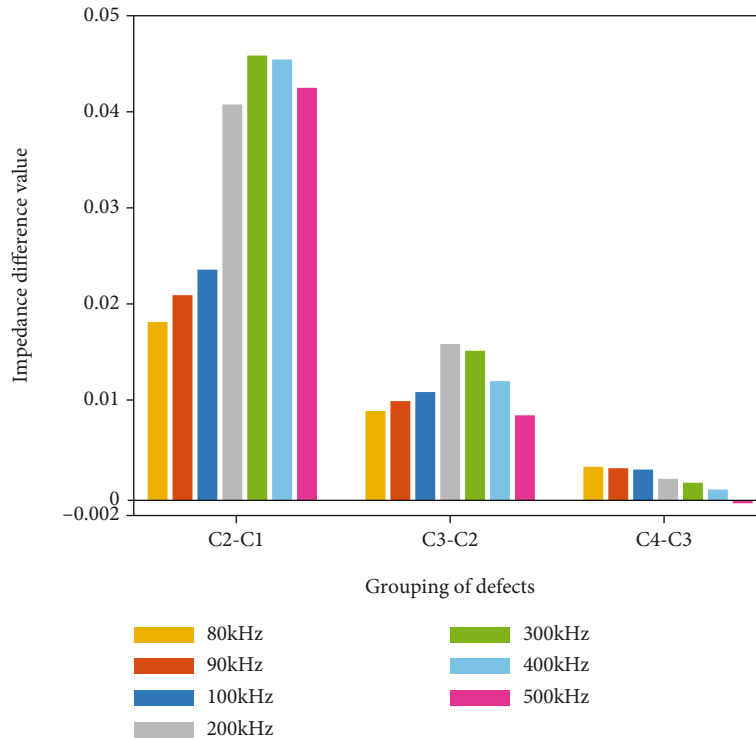
FIGURE 12: Influence of different excitation frequencies on impedance difference of surface defects at different depths.
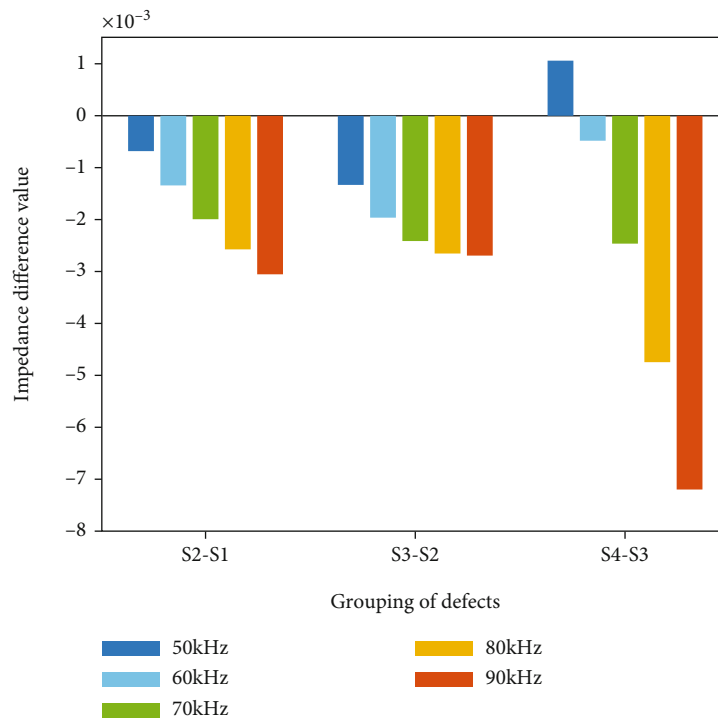
FIGURE 13: Influence of different excitation frequencies on impedance difference of subsurface defects with different depths.

impedance difference gradually decreases. The impedance difference Z_{diff} of defects S1–S4 is 0 at different excitation frequencies; that is, the impedance value at the defect is equal to that at the nondefect. When the excitation frequency is increased, the impedance value at the defect is less than that of the nondefect, and the impedance difference Z_{diff} is negative.

According to the previous analysis, in the ECT process, the greater the difference in signals between different defects, the more conducive to the identification, classification, and

Table 5: Simulation and analysis results of parameter setting.

Parameter		Value
Probe type		Absolute probe
Outer radius of coil (r_2)		3.3 mm
Inner radius of coil (r_1)		$\leq 0.6 \times r_2$
Excitation frequency	Surface defect	80~400 kHz
	Subsurface defect	60~90 kHz

quantitative analysis of defects. To further determine the optimal excitation frequency for defect detection of aviation aluminum alloy plates, the difference in impedance between adjacent defects was calculated. The calculated differences between surface defects and subsurface defects are shown in Figures 12 and 13, respectively.

In the statistics of impedance difference of surface defects shown in Figure 12, the impedance difference of adjacent defects under six frequency excitations of 80 kHz, 90 kHz, 100 kHz, 200 kHz, 300 kHz, 400 kHz, and 500 kHz was analyzed. It can be observed that as the defect depth increases, the impedance difference corresponding to each excitation frequency decreases, and the impedance value of deeper defects is less than that of shallow defects at higher excitation frequencies. Combined with the analysis of Figures 10 and 12, for the surface defects of the aviation aluminum alloy plate, the excitation frequency can achieve an accurate detection effect in the range of 80 kHz to 400 kHz.

Similarly, in order to accurately classify, identify, and quantitatively analyze a variety of subsurface defects, the impedance difference of adjacent defects was analyzed from six excitation frequencies of 50 kHz, 60 kHz, 70 kHz, 80 kHz, and 90 kHz, as shown in Figure 13. In shallow subsurface defects, such as 1 mm, 2 mm, and 3 mm deep defects, the impedance difference increases with an increase in the excitation frequency. However, when the defect depth is 3–4 mm and the excitation frequency is 50 kHz, the impedance value of shallow defects is greater than that of deep defects, which further causes difficulty in defect classification and identification as well as quantitative evaluation.

Through comprehensive analysis of the optimal excitation frequency of defect detection shown in Figures 12 and 13, surface defects and subsurface defects can achieve good detection results in the excitation frequency range of 80 kHz~90 kHz. In the actual defect detection process, the 80 kHz~90 kHz excitation frequency can be used to preliminarily judge the defect type and then adjust the excitation frequency for different types of defects to collect defect signals to realize accurate identification, classification, and quantitative analysis of defects. The simulation analysis results of the ECT parameters for aviation aluminum alloy plate defects are listed in Table 5.

When ECT technology was used for defect detection of aviation aluminum alloy plates, an absolute probe with an outer radius of 3.3 mm was used as the probe, and the inner radius of the coil was less than 0.6 times its outer radius. At this time, the probe can simultaneously meet the accuracy of

defect detection and edge recognition. In addition, for the setting of the excitation frequency during defect detection, the frequency adaptation ranges of surface and subsurface defects are 80~400 kHz and 60~90 kHz, respectively. An excitation frequency of 80~90 kHz can be used for detection to realize accurate detection, identification, and quantitative analysis of defects.

5. Conclusion

To address the problem of surface and subsurface defect detection as well as quantitative analysis of aviation aluminum alloy plates, a comprehensive simulation optimization analysis was conducted using probe type, coil size, and excitation frequency. Compared with the differential probe, the absolute probe has better defect identifiability and defect edge detection ability. Meanwhile, when the outer radius is 3.3 mm, the absolute probe can further improve the defect detection ability and defect edge detection accuracy. In addition, for the excitation frequency range of 80 kHz to 90 kHz, the detection ability for unknown defects was further improved. The optimization analysis of the parameter settings of defect detection through simulation modeling can guide the probe design and excitation frequency setting in the actual detection process. In addition, it can promote the application of computer-aided design and optimization in the eddy current detection of aviation aluminum alloy defects. The future work of this paper will continue from two aspects: First, we will combine simulation signals and intelligent algorithms to further improve the accuracy of defect edge recognition; thereafter, we will verify and improve it with the measured signal. Second, we will combine the simulation model and the measured signal to study the influence of different scanning paths on defect detection.

Data Availability

The [data.rar] data used to support the findings of this study have been deposited in the Aliyun Server repository (http://120.76.226.240/KH_AnCore/data.rar), and other researchers can download original data from this URL.

Conflicts of Interest

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

References

[1] A. Sophian, G. Y. Tian, D. Taylor, and J. Rudlin, "Design of a pulsed eddy current sensor for detection of defects in aircraft lap-joints," *Sensors and Actuators A: Physical*, vol. 101, no. 1-2, pp. 92–98, 2002.

[2] T. Chen, G. Y. Tian, A. Sophian, and P. W. Que, "Feature extraction and selection for defect classification of pulsed eddy current Ndt," *Ndt & E International*, vol. 41, no. 6, pp. 467–476, 2008.

[3] M. Morozov, G. Y. Tian, and P. J. Withers, "The pulsed eddy current response to applied loading of various aluminium alloys," *NDT & E International*, vol. 43, no. 6, pp. 493–500, 2010.

[4] X. Chen, D. Hou, L. Zhao, P. Huang, and G. Zhang, "Study on defect classification in multi-layer structures based on fisher linear discriminate analysis by using pulsed eddy current technique," *NDT & E International*, vol. 67, pp. 46–54, 2014.

[5] W. Deng, B. Ye, J. Bao, G. Huang, and J. Wu, "Classification and quantitative evaluation of eddy current based on kernel-Pca and elm for defects in metal component," *Metals*, vol. 9, no. 2, p. 155, 2019.

[6] J. Kim, G. Yang, L. Udpa, and S. Udpa, "Classification of pulsed eddy current Gmr data on aircraft structures," *Ndt & E International*, vol. 43, no. 2, pp. 141–144, 2010.

[7] S. Hosseini and A. A. Lakis, "Application of time–frequency analysis for automatic hidden corrosion detection in a multi-layer aluminum structure using pulsed eddy current," *Ndt & E International*, vol. 47, pp. 70–79, 2012.

[8] Y. He, M. Pan, D. Chen, and F. Luo, "Pec defect automated classification in aircraft multi-ply structures with interlayer gaps and lift-offs," *NDT & E International*, vol. 53, pp. 39–46, 2013.

[9] Y. Li, B. Yan, D. Li, H. Jing, Y. Li, and Z. Chen, "Pulse-modulation eddy current inspection of subsurface corrosion in conductive structures," *Ndt & E International*, vol. 79, pp. 142–149, 2016.

[10] M. R. Perumal, K. Balasubramaniam, and K. Arunachalam, "Study of the choice of excitation frequency for sub surface defect detection in electrically thick conducting specimen using eddy current testing," *Journal of Nondestructive Evaluation*, vol. 37, no. 4, p. 70, 2018.

[11] B. Yan, Y. Li, S. Ren, I. M. Zainal Abidin, Z. Chen, and Y. Wang, "Recognition and evaluation of corrosion profile via pulse-modulation eddy current inspection in conjunction with improved Canny algorithm," *NDT & E International*, vol. 106, pp. 18–28, 2019.

[12] Y. Xie, J. Li, Y. Tao, S. Wang, W. Yin, and L. Xu, "Edge effect analysis and edge defect detection of titanium alloy based on eddy current testing," *Applied Sciences*, vol. 10, no. 24, p. 8796, 2020.

[13] Y. He, F. Luo, M. Pan, X. Hu, J. Gao, and B. Liu, "Defect classification based on rectangular pulsed eddy current sensor in different directions," *Sensors and Actuators A: Physical*, vol. 157, no. 1, pp. 26–31, 2010.

[14] J. Wu, D. Zhou, and J. Wang, "Surface crack detection for carbon fiber reinforced plastic materials using pulsed eddy current based on rectangular differential probe," *Journal of Sensors*, vol. 2014, 8 pages, 2014.

[15] K. Zhang, Y. He, and Z. Dong, "Pulsed eddy current nondestructive testing for defect evaluation and imaging of automotive lightweight alloy materials," *Journal of Sensors*, vol. 2018, 11 pages, 2018.

[16] D. J. Pasadas, H. G. Ramos, B. Feng, P. Baskaran, and A. L. Ribeiro, "Defect classification with Svm and wideband excitation in multilayer aluminum plates," *IEEE Transactions on Instrumentation and Measurement*, vol. 69, no. 1, pp. 241–248, 2020.

[17] H. Wang, W. Li, and Z. Feng, "Noncontact thickness measurement of metal films using eddy-current sensors immune to distance variation," *IEEE Transactions on Instrumentation and Measurement*, vol. 64, no. 9, pp. 2557–2564, 2015.

Hindawi

Research Article

Calibration of Collaborative Robots Based on Position Information and Local Product of Exponentials

Guanbin Gao ⓘ, Yangtao Gao ⓘ, Fei Liu ⓘ, and Jing Na ⓘ

Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China

Correspondence should be addressed to Fei Liu; feiliu2017@foxmail.com

To improve the positioning accuracy of collaborative robots, a novel modeling and calibration method for collaborative robots is proposed based on position information and modified local product of exponentials (LPoE). The kinematic error model is derived from the kinematic model through differential transformation. To solve the problem of the high redundancy and complexity of the error model that is difficult to identify, the errors of the kinematic parameters are all transferred to the initial position and posture matrix of the local coordinate system, which simplifies the error model and improves the speed and accuracy of the identification calculation. However, the simplified error model still requires posture data which are very difficult to acquire in practice. For this reason, the position error is separated from the kinematic model, and an error model based on position data is established. Then, a kinematic calibration method of the collaborative robot based on position data is proposed, which simplifies the measurement process and improves the efficiency of calibration. The effectiveness of the method is verified by simulations and experiments on a six degree-of-freedom collaborative robot.

1. Introduction

With the development of robots and the change of manufacturing mode, the working field of robots is expanding to realize natural interaction with humans, e.g., collaborative robots begin to be applied in unstructured environments to cooperate with humans. The collaborative robot is a new type of industrial robots, which can directly interact with people in the designated cooperative area [1]. Compared with traditional industrial robots, collaborative robots have the advantages of high security, universality, sensitivity, easy to use, and man-machine cooperation, which can be applied in the field of industrial production, family services, and rehabilitation medicine [2–4]. Generally, the actual kinematic parameters of the robot are inconsistent with the nominal parameters in its controller for the error of manufacturing and assembly and the stress deformation of joints and linkages, which results in control errors and reduces the positioning accuracy of the robot. Moreover, the collaborative robot is an open-chain mechanism with low stiffness and error accumulation and amplification. Therefore, the accuracy of the collaborative robot is much

lower than that of traditional orthogonal instruments, and it cannot be applied to cases with high accuracy requirements, such as measuring, precision assembly, turning, and milling [5, 6].

Kinematic calibration is one of the main methods to improve the absolute positioning accuracy of robots, which is the process of identifying the actual kinematic parameters of the robots by using advanced measurement means and model-based parameter identification method, and compensating them to the controller of robots or compensating to improve positioning accuracy [7]. Kinematic calibration has been a research hotspot in the field of robots [8–10], which consists of four steps: kinematic modeling, pose measuring, parameter identification, and compensation [11]. Most kinematic models in the controller of robots are based on Denavit-Hartenberg (D-H) method [12]. However, when two adjacent axes of the robot are parallel, the Jacobian matrix of the D-H model will be singular [13]; that is, the kinematic solution has abrupt changes and discontinuities, which results in coupling problem between some of the kinematic parameters [14]. To solve the singularity, Park et al. [15] proposed a kinematic modeling method, i.e.,

PoE, which only needs to build the base and tool coordinate systems on the 0th linkage and the end. Nevertheless, the PoE error model contains a finite integral term, which is not an explicit form and is inconvenient to use leading to the identifiability of its kinematic parameters cannot be determined [16]. He et al. [17] modified the PoE model by deriving the differential solution formula of exponential mapping on joint twists and variables and established an error model based on PoE model with explicit expression, which is the main PoE method used in kinematic calibration at present. Jiang et al. [18] used the PoE model to analyze the position space relationship of joints' axis under different geometric errors and proposed a calibration model including joint errors. Wang et al. [19] proposed a self-kinematic system for an on-orbit space manipulator based on PoE formula, which can measure the position and posture of the manipulator without using external measurement devices.

In the kinematic parameter identification of robots, the D-H model only needs the end position data of robots, while the traditional PoE model also needs to obtain the position and posture data of robots. It is difficult to obtain posture data in practice. The operation is cumbersome, time-consuming, and difficult to ensure its accuracy [20]. Therefore, the traditional PoE model is difficult to be applied in the actual kinematic calibration of robots.

There are $13 \times n$ parameters in the traditional PoE-based error model which is very difficult to be identified. In this paper, to simplify the error model and improve the redundancy, the error model is simplified by transferring all the errors of the kinematic parameters to the initial position and posture matrix, to avoid the use of posture data which is difficult to obtain. The position error is separated from the kinematic model, and an error model based on position data is established. Then, a kinematic calibration method of collaborative robot based on position data is proposed, which simplifies the measurement process and improves the efficiency of calibration.

2. Kinematic Modeling based on LPoE for the Collaborative Robot

In this section, the transformation of the local coordinate system of the adjacent linkages is derived, based on which the LPoE kinematic model of a typical six degree-of-freedom collaborative robot is established.

2.1. Description of the Coordinate System of Adjacent Linkages. Figure 1 shows a typical six degree-of-freedom collaborative robot. Unlike general industrial robots, the axes of the second, third, and fourth joints of the collaborative robot are parallel to each other.

Like the D-H model, the LPoE model uses a local coordinate system to describe the motion relationship of adjacent links [21]. As shown in Figure 2, the relationship between the adjacent linkages $i-1$ and i can be written as (1).

$$T_{i-1,i}(q_i) = T_{i-1,i}(0)e^{\hat{s}_i q_i}, \tag{1}$$

where $T_{i-1,i}(0) \in SE(3)$ is the initial transformation matrix

FIGURE 1: The six degree-of-freedom collaborative robot.
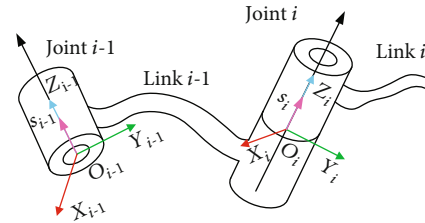
FIGURE 2: The coordinate systems of adjacent joints.

between coordinate systems of i and $i-1$, $\hat{s}_i \in so(3)$ denotes the twist of joint i, q_i denotes the rotation angle of joint i, and $e^{\hat{s}_i q_i}$ is a matrix:

$$e^{\hat{s}_i q_i} = \begin{bmatrix} e^{\hat{w}_i q_i} & v_i q_i \\ 0 & 1 \end{bmatrix}, \tag{2}$$

where $e^{\hat{w}_i q_i} = I + \hat{w}_i \sin q_i + \hat{w}_i^2 (1 - \cos q_i)$.

Furthermore, $\hat{s}_i$ can be written in the coordinate system i:

$$\hat{s}_i = \begin{bmatrix} \hat{w}_i & v_i \\ 0 & 0 \end{bmatrix}, \tag{3}$$

where $v_i = [v_{ix}, v_{iy}, v_{iz}]^T \in \mathbb{R}^{3 \times 1}$ and $\hat{w}_i \in so(3)$ are the cross-product matrixes of $w_i = [w_{ix}, w_{iy}, w_{iz}] \in \mathbb{R}^{3 \times 1}$. $\hat{w}_i$ can be

written as

$$\hat{w}_i = \begin{bmatrix} 0 & -w_{iz} & w_{iy} \\ w_{iz} & 0 & -w_{ix} \\ -w_{iy} & w_{ix} & 0 \end{bmatrix}. \quad (4)$$

To be consistent with existing robot position and posture description methods, $T_{i-1,i}(0) \in SE(3)$ can be written as the form of position and posture matrix:

$$T_{i-1,i}(0) = \begin{bmatrix} R_{i-1,i}(0) & d_{i-1,i}(0) \\ 0 & 1 \end{bmatrix}, \quad 7 \quad (5)$$

where $R_{i-1,i}(0) \in SO(3)$ and $d_{i-1,i}(0) \in \mathbb{R}^{3\times1}$ denote the initial posture and position, respectively.

2.2. Kinematic Modeling for the Robot. Based on the coordinate systems of adjacent joints, the kinematic model of the six degree-of-freedom collaborative robot can be derived as follows:

$$T_{0,6}(q_1, q_2, \cdots, q_6) = T_{0,1}(q_1)T_{1,2}(q_2) \cdots T_{5,6}(q_6)$$
$$= T_{0,1}(0)e^{\hat{s}_1 q_1}T_{1,2}(0)e^{\hat{s}_2 q_2} \cdots T_{5,6}(0)e^{\hat{s}_6 q_6}. \quad (6)$$

$T_{6,7}$ is used to denote the transformation from the flange to the tool of the robot. Then, the position and posture matrix of the tool will be

$$T_{0,7}(q_1, q_2, \cdots, q_6) = T_{0,1}(0)e^{\hat{s}_1 q_1}T_{1,2}(0)e^{\hat{s}_2 q_2} \cdots T_{5,6}(0)e^{\hat{s}_6 q_6}T_{6,7}. \quad (7)$$

Following the method of LPoE, the coordinate systems of the robot were established as shown in Figure 3. The nominal value of the twists is $s_1 = s_2 = s_3 = s_4 = s_5 = s_6 = [0\,0\,1\,0\,0\,0]^T$. The nominal kinematic parameters can be determined according to the structural size of the robot, as shown in Table 1.

3. The Error Modeling Based on LPoE for the Collaborative Robot

In this section, the error modeling of the collaborative robot based on LPoE is proposed and simplified. Then, the pose-based and position-based error models are derived, respectively. The identification method and flow chart based on the least square method are presented.

3.1. Error Modeling and Simplifying. The kinematic model, i.e., (7) can be written as the function of $T(0)$, s, and q:

$$T = f(T(0), s, q), \quad (8)$$

where $T(0) = [T_{0,1}(0), T_{1,2}(0), \cdots, T_{6,7}(0)]^T$, $s = [s_1, s_2, \cdots, s_6]^T$, and $q = [q_1, q_2, \cdots, q_6]^T$.
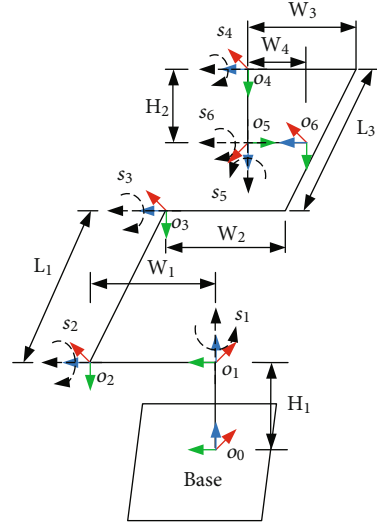
FIGURE 3: The coordinate systems and twists of the robot.

TABLE 1: The kinematic parameters of the robot.

Name	H_1	W_1	L_1	W_2	L_2	W_3	H_2	W_4
Value [mm]	96	138	418	114	398	98	98	89

By linearization of (8), the error model can be derived:

$$\delta T T^{-1} = \left(\frac{\partial f}{\partial T(0)}\delta T(0) + \frac{\partial f}{\partial s}\delta s + \frac{\partial f}{\partial q}\delta q \right)T^{-1}, \quad (9)$$

where $\delta T T^{-1}$ denotes the error of position and posture of the tool in the base coordinate system, which can be calculated by subtracting the measuring position and posture from the nominal ones. Kinematic parameters can be obtained by identification with the measuring data and the error model. The objective function of the identification model is generally set as the least square solution, i.e.,

$$\text{Min} \left(\sum \left\| \delta T T^{-1} - \left(\frac{\partial f}{\partial T(0)}\delta T(0) + \frac{\partial f}{\partial s}\delta s + \frac{\partial f}{\partial q}\delta q \right)T^{-1} \right\| \right). \quad (10)$$

There are 13 parameters to be identified in each joint in (10), and there are 6 joints in the collaborative robot. So, many parameters make the parameter identification model highly redundant and complicated. It is very difficult to directly identify the parameters.

To solve the problem of high redundancy and complexity of the error model based on LPoE, the simplified error model will be presented. As shown in Figure 4, the nominal coordinate system of joint i $\{o_i x_i y_i z_i\}$ and the actual coordinate system $\{o_i^a x_i^a y_i^a z_i^a\}$ are not coincident for the error of kinematic parameters, i.e., the nominals $T_{i-1,i}(0)$, s_i and q_i are not equal to the actual $T^a_{i-1,i}(0)$, s_i^a, and q_i^a. For the local coordinate system of LPoE can be assigned arbitrarily, the following two assumptions are proposed [21]: (1) the kinematic error only exists in the initial position and posture
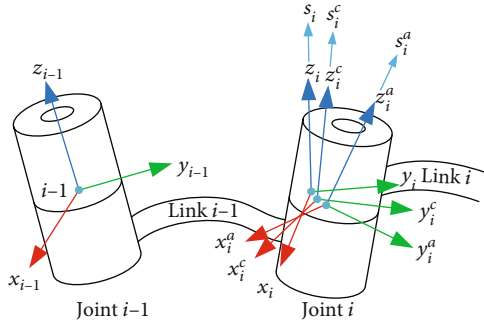
FIGURE 4: The error of kinematic parameters in coordinate systems.

of the local coordinate system $T(0)$, and (2) the nominal values of q_i and s_i remain unchanged in the whole parameter identification calculation process.

Based on these assumptions, a new local coordinate system $T_i^c(0) = \{o_i^c x_i^c y_i^c z_i^c\}$ is set at each joint as the intermedia coordinate system in the following identification, as shown in Figure 4. Then, q_i^c and s_i^c will remain unchanged as q_i and s_i in the identification. Therefore, (9) and (10) can be simplified as

$$\delta T T^{-1} = \frac{\partial f}{\partial T(0)} \delta T(0) T^{-1},$$

$$\text{Min}\left(\sum \left\| \delta T T^{-1} - \frac{\partial f}{\partial T(0)} \delta T(0) T^{-1} \right\|\right).$$

(11)

3.2. Error Modeling Based on Position and Posture. According to Chasles theorem [22], for a given homogeneous transformation matrix $T \in SE(3)$, there is at least one coordinate twist $\hat{p} \in se(3)$ such that $e^{\hat{p}} = T$. Then, one can obtain (12) according to (1).

$$e^{\hat{p}_i} = T_{i-1,i}(0),$$

(12)

where $\hat{p}_i \in se(3)(i = 1, 2, \cdots, 7)$. Then, (7) can be rewritten as

$$T_{0,7} = e^{\hat{p}_1} e^{\hat{s}_1 q_1} e^{\hat{p}_2} e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7}.$$

(13)

Let $\delta \hat{p}_i$ denote the error of $\hat{p}_i$. Then, one can obtain $\delta \hat{p}_i = e^{-\hat{p}_i} \delta(e^{\hat{p}_i})$ and $\delta(e^{\hat{p}_i}) = e^{\hat{p}_i} \delta \hat{p}_i$. By linearization of (13), one can get

$$\begin{aligned}
\delta T_{0,7} &= \delta\left(e^{\hat{p}_1}\right) e^{\hat{s}_1 q_1} e^{\hat{p}_2} e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7} \\
&\quad + e^{\hat{p}_1} e^{\hat{s}_1 q_1} \delta\left(e^{\hat{p}_2}\right) e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7} + \cdots \\
&\quad + e^{\hat{p}_1} e^{\hat{s}_1 q_1} \cdots e^{\hat{p}_5} e^{\hat{s}_5 q_5} \delta\left(e^{\hat{p}_6}\right) e^{\hat{s}_6 q_6} e^{\hat{p}_7} \\
&\quad + e^{\hat{p}_1} e^{\hat{s}_1 q_1} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} \delta\left(e^{\hat{p}_7}\right) \\
&= e^{\hat{p}_1} \delta \hat{p}_1 e^{\hat{s}_1 q_1} e^{\hat{p}_2} e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7} \\
&\quad + e^{\hat{p}_1} e^{\hat{s}_1 q_1} e^{\hat{p}_2} \delta \hat{p}_2 e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7} \\
&\quad + \cdots + e^{\hat{p}_1} e^{\hat{s}_1 q_1} \cdots e^{\hat{p}_5} e^{\hat{s}_5 q_5} e^{\hat{p}_6} \delta \hat{p}_6 e^{\hat{s}_6 q_6} e^{\hat{p}_7} \\
&\quad + e^{\hat{p}_1} e^{\hat{s}_1 q_1} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6} e^{\hat{p}_7} \delta \hat{p}_7.
\end{aligned}$$

(14)
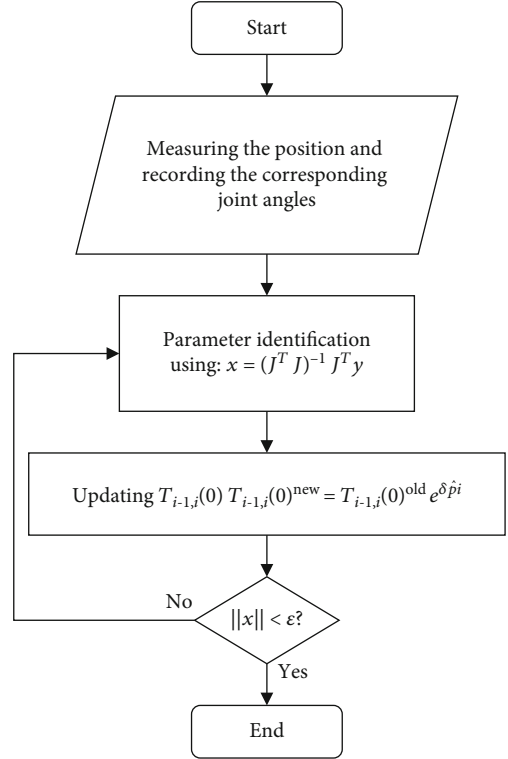
FIGURE 5: The flow chart of kinematic parameters identification by the iterative least square method.

Right multiplying $T_{0,7}^{-1}$ both sides of (14), one can get (15) in the form of adjoint transformation.

$$\begin{aligned}
\delta T_{0,7} T_{0,7}^{-1} &= \text{Ad}_{e^{\hat{p}_1}} \delta \hat{p}_1 + \text{Ad}_{e^{\hat{p}_1} e^{\hat{s}_1 q_1}} \text{Ad}_{e^{\hat{p}_2}} \delta \hat{p}_2 + \cdots \\
&\quad + \text{Ad}_{e^{\hat{p}_1} e^{\hat{s}_1 q_1} e^{\hat{p}_2} e^{\hat{s}_2 q_2} \cdots e^{\hat{p}_6} e^{\hat{s}_6 q_6}} \text{Ad}_{e^{\hat{p}_7}} \delta \hat{p}_7,
\end{aligned}$$

(15)

where $\delta T_{0,7} T_{0,7}^{-1} \in SE(3)$ denotes the total error of kinematic parameters of the robot relative to the base coordinate system. Ad is the adjoint transformation defined as

$$\text{Ad} : SE(3) \mapsto \mathfrak{R}^{6 \times 6}, g = \begin{bmatrix} R & P \\ 0 & 1 \end{bmatrix} \mapsto \text{Ad}_g = \begin{bmatrix} R & \hat{P}R \\ 0 & R \end{bmatrix},$$

(16)

where g is a homogeneous transformation matrix, R is the orthogonal matrix in g, and $\hat{P}$ is the antisymmetric matrix of P. According to the matrix logarithm calculation formula defined in SE (3), $\delta T_{0,7} T_{0,7}^{-1}$ can be written as

$$\delta T_{0,7} T_{0,7}^{-1} = \log\left(T_{0,7}^a T_{0,7}^{-1}\right),$$

(17)

where $T_{0,7}^a$ denotes the actual position and posture. Components of SE(3) can be expressed by $\delta \hat{p}_i \mapsto \delta p_i \in \mathfrak{R}^{6 \times 1}$ and $\log\left(T_{0,7}^a T_{0,7}^{-1}\right) \mapsto \log\left(T_{0,7}^a T_{0,7}^{-1}\right)^\vee \in \mathfrak{R}^{6 \times 1}$. Then, (15) can be

TABLE 2: The setting error of the robot.

Joint	$\delta p_i{'}$	$\delta s_i{'}$	$\delta q_i{'}$
0-1	$(0, 0.001, 0.002, 2, 1.5, 0.5)^T$	$(0, 0, 0, 0, \sin(0.01), -1 + \cos(0.01))^T$	0
1-2	$(0, 0.002, 0.003, -2, 1.2, 2.4)^T$	$(0, 0, 0, 0, -\sin(0.015), -1 + \cos(0.015))^T$	0.03
2-3	$(0.001, 0.004, 0.003, 2, -1, 2)^T$	$(0, 0, 0, \sin(0.023), 0, -1 + \cos(0.023))^T$	0.01
3-4	$(0.001, 0.005, 0.002, 2, 1.5, -1)^T$	$(0, 0, 0, -\sin(0.004), 0, -1 + \cos(0.004))^T$	0.02
4-5	$(0.002, 0.003, 0, -1, 1.6, 2.3)^T$	$(0, 0, 0, 0, \sin(0.031), -1 + \cos(0.031))^T$	0.015
5-6	$(0.003, 0.001, -0.004, 2, 3, 1)^T$	$(0, 0, 0, \sin(0.013), 0, -1 + \cos(0.013))^T$	0.02
6-7	$(0.005, 0.002, -0.003, 1.6, 2, 3.2)^T$	—	—
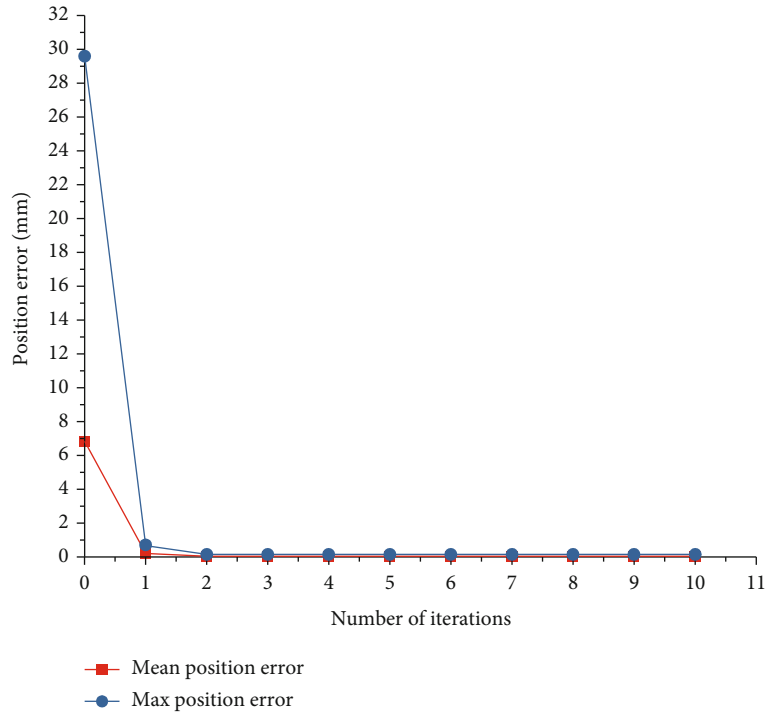
FIGURE 6: The process of iterative convergence.

rewritten as

$$\log\left(T_{0,7}^{a} T_{0,7}^{-1}\right)^{\vee} = \mathrm{Ad}_{T_{0,1}(0)}\delta p_1 + \mathrm{Ad}_{T_{0,1}}\mathrm{Ad}_{T_{1,2}(0)}\delta p_2 + \cdots + \mathrm{Ad}_{T_{0,6}}\mathrm{Ad}_{T_{6,7}(0)}\delta p_7, \quad (18)$$

where $T_{0,7} = e^{\hat{p}_1}e^{\hat{s}_1 q_1}e^{\hat{p}_2}e^{\hat{s}_2 q_2}\cdots e^{\hat{p}_6}e^{\hat{s}_6 q_6}e^{\hat{p}_7}$ denotes the transformation from coordinate system 0th to 7th. The initial position and posture transformation of linkage i relatively to $i-1$ is

$$T_{i-1,i}^{c}(0) = e^{\hat{p}_i}e^{\delta\hat{p}_i} = T_{i-1,i}(0)e^{\delta\hat{p}_i}. \quad (19)$$

(18) can be written in the form of linear equations:

$$y = Ax, \quad (20)$$

where

$$y = \log\left(T_{0,7}^{a} T_{0,7}^{-1}\right)^{\vee} \in \mathfrak{R}^{6\times 1},$$

$$x = [\delta p_1, \delta p_2, \cdots, \delta p_7]^{\mathrm{T}} \in \mathfrak{R}^{6(7)\times 1},$$

$$A = \left[\mathrm{Ad}_{T_{0,1}(0)}, \mathrm{Ad}_{T_{0,1}}\mathrm{Ad}_{T_{1,2}(0)}, \cdots, \mathrm{Ad}_{T_{0,6}}\mathrm{Ad}_{T_{6,7}(0)}\right] \in \mathfrak{R}^{6\times 6\times 7}. \quad (21)$$

$T_{0,7}^{-1}$ and A can be obtained from the nominal model. $T_{0,7}^{a}$ can be obtained by measuring, usually a laser tracker. x denotes the error of the kinematic parameters to be identified. To improve the accuracy and robustness of the identification, it is usually necessary to measure the position and posture of the tool under different configurations of the robot. If m times of measuring are required, at the i -$^{\mathrm{th}}$

(a) The position error in the x direction

(b) The position error in the y direction

(c) The position error in the z direction
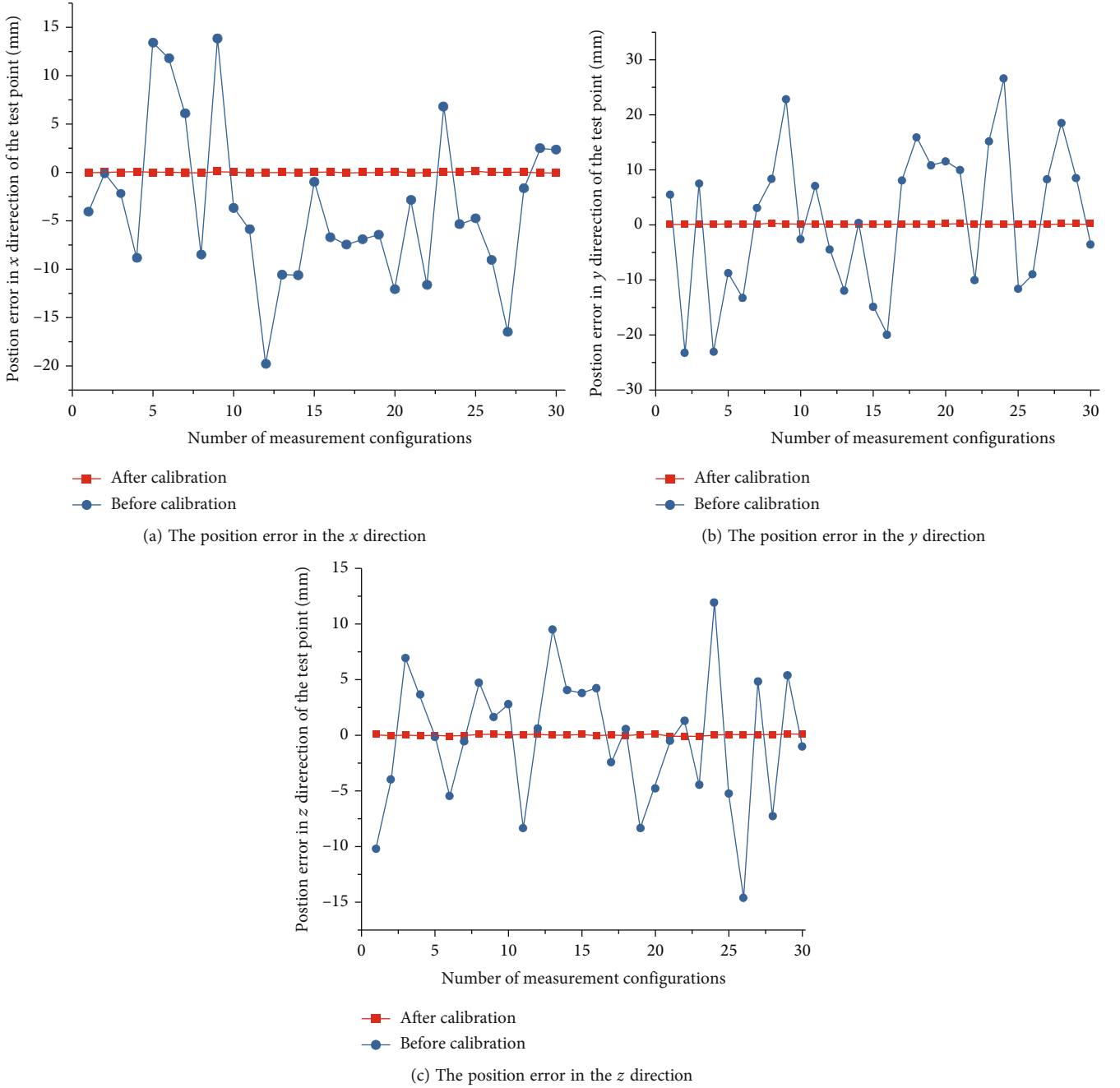
FIGURE 7: The position error in directions of x, y, and z before and after calibration in simulations.

time, the position and posture error y_i and the corresponding adjoint mapping matrix A_i can be obtained. After finishing acquiring m measuring data, y_i and A_i are substituted into (22).

$$Y = \begin{bmatrix} y^1 \\ \vdots \\ y^m \end{bmatrix} = \begin{bmatrix} A^1 \\ \vdots \\ A^m \end{bmatrix} x = Ax, \tag{22}$$

where

$$\begin{aligned} Y &= [y_1, y_2, \cdots, y_m]^{\mathrm{T}} \in \mathfrak{R}^{6m \times 1}, \\ x &= [\delta p_1, \delta p_2, \cdots, \delta p_7]^{\mathrm{T}} \in \mathfrak{R}^{6 \times 7 \times 1}, \\ A &= [A_1, A_2, \cdots, A_m] \in \mathfrak{R}^{6m \times 6 \times 7}. \end{aligned} \tag{23}$$

Since the model consists of $6 \times m$ linear equations and variables, when $m > 7$, the linear least squares algorithm can be used for parameter identification, and the least

squares solution of x is

$$x = \left(A^{T}A\right)^{-1}A^{T}Y, \tag{24}$$

where $\left(A^{T}A\right)^{-1}A^{T}$ is the pseudoinverse of A.

$T(0)$ can be initialized with the nominal value, i.e., $T(0) = [T_{0,1}(0), T_{1,2}(0), \cdots, T_{6,7}(0)]^{T}$. x can be calculated by acquired data and (24). Then, $T(0)$ can be updated by (19) till the objective function approaches 0 or $T(0)$ converges to a stable value. Then, the stable $T(0)$ is set as the initial position and posture, i.e., $T(0) = [T_{0,1}(0)^{c}, T_{1,2}(0)^{c}, \cdots, T_{6,7}(0)^{c}]^{T}$. (7) can be written as (25) after the nominal position and posture matrix is substituted with the above identified one.

$$T_{0,7} = T_{0,1}^{c}(0)e^{\widehat{s}_{1}q_{1}}T_{1,2}^{c}(0)e^{\widehat{s}_{2}q_{2}} \cdots T_{5,6}^{c}(0)e^{\widehat{s}_{6}q_{6}}T_{6,7}^{c}. \tag{25}$$

3.3. Error Modeling Based on Position.

In practice, the posture of the robot is very difficult to obtain, which is time consuming and complicated. One error modeling method based on position is presented in this paper.

Right multiplying P_{0} both sides of (13), one can get

$$\begin{bmatrix} P \\ 1 \end{bmatrix} = T_{0,7}P_{0} = e^{p_{1}}e^{\widehat{s}_{1}q_{1}}e^{p_{2}}e^{\widehat{s}_{2}q_{2}} \cdots e^{p_{6}}e^{\widehat{s}_{6}q_{6}}e^{p_{7}}P_{0}, \tag{26}$$

where P denotes the position of the tool, $P_{0} = [0,0,0,1]^{T}$. Then, the position error model can be obtained by differentiating (26):

$$\begin{bmatrix} \delta P \\ 0 \end{bmatrix} = \delta T_{0,7} \begin{bmatrix} P_{0} \\ 1 \end{bmatrix} + T_{0,7} \begin{bmatrix} \delta P_{0} \\ 0 \end{bmatrix}. \tag{27}$$

According to the assumptions above, i.e., the kinematic error only exists in the initial position and posture of the local coordinate system $T(0)$, (27) can be written as

$$\begin{aligned} \begin{bmatrix} \delta P \\ 0 \end{bmatrix} &= \delta T_{0,7} \begin{bmatrix} P_{0} \\ 1 \end{bmatrix} = \left(\delta T_{0,7}T_{0,7}^{-1}\right)T_{0,7}\begin{bmatrix} P_{0} \\ 1 \end{bmatrix} \\ &= \left(\delta T_{0,7}T_{0,7}^{-1}\right)\begin{bmatrix} P \\ 1 \end{bmatrix} = \begin{bmatrix} -\widehat{P} & I_{3} \\ 0 & 0 \end{bmatrix}\left(\delta T_{0,7}T_{0,7}^{-1}\right)^{\vee}. \end{aligned} \tag{28}$$

By substituting (18) into (28), one can get

$$\delta P = \left[-\widehat{P}I_{3}\right]\left(Ad_{T_{0,1}(0)}\delta p_{1} + Ad_{T_{0,1}}Ad_{T_{1,2}(0)}\delta p_{2} + \cdots + Ad_{T_{0,6}}Ad_{T_{6,7}(0)}\delta p_{7}\right). \tag{29}$$

(29) can be simplified as

$$y = Jx, \tag{30}$$

where y $= \delta P = P_{a} - P_{n} \in \mathfrak{R}^{3 \times 1}$, P_{n} denotes the nominal position, P_{a} denotes the actual position, $x = [\delta p_{1}, \delta p_{2}, \cdots \delta p_{7}]^{T} \in$
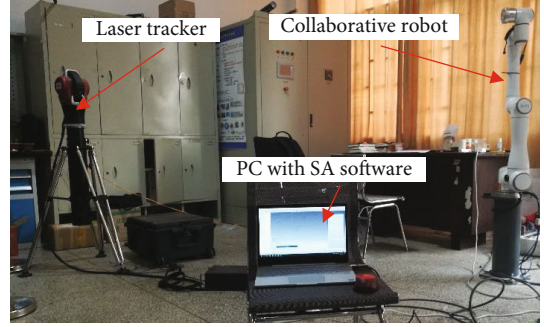
FIGURE 8: The experimental system of calibration for the collaborative robot.

$\mathfrak{R}^{6 \times 7 \times 1}$, and $J = \begin{bmatrix} -\widehat{P} & I_{3} \end{bmatrix}[Ad_{T_{0,1}(0)}, Ad_{T_{0,1}}Ad_{T_{1,2}(0)}, \cdots Ad_{T_{0,6}}$ $Ad_{T_{6,7}(0)}] \in \mathfrak{R}^{6 \times 6 \times 7}$.

The position of the tool can be measured m times at different configurations. Then, with these measured data, the kinematic parameter identification equations can be obtained as shown in (31).

$$Y = \begin{bmatrix} y^{1} \\ \vdots \\ y^{m} \end{bmatrix} = \begin{bmatrix} J^{1} \\ \vdots \\ J^{m} \end{bmatrix}x = Jx. \tag{31}$$

One can solve (31) by the iterative least square method with the termination condition of the iteration as $\|x\| < \varepsilon$, as shown in Figure 5. And the least square solution is

$$x = \left(J^{T}J\right)^{-1}J^{T}Y. \tag{32}$$

4. Simulations

In the simulation, the twists are set as Figure 3, and the kinematic parameters are set as Table 1. To simulate the measuring error and test the stability of the error model, Gaussian noise with an expectation of 0 and a variance of 0.2 is applied to the data of positions calculated by the kinematic model of the robot. The settings when calculated the data of positions are the following: (1) the errors of kinematic parameters are $\delta p_{i}{}'$, $\delta q_{i}{}'$, and $\delta s_{i}{}'$, and (2) the actual initial position and posture is set as follows:

$$T_{i-1,i}^{a}(0) = T_{i-1,i}(0)e^{\delta \widehat{p}_{i}{}'} \ (i = 1, 2, \cdots, 7). \tag{33}$$

The actual position is calculated by

$$P_{0,7}^{a} = \prod_{i=1}^{6}\left(T_{i-1,i}^{a}(0)e^{\left(\widehat{s}_{i} + \delta \widehat{s}_{i}{}'\right)\left(q_{i} + \delta q_{i}{}'\right)}\right)T_{6,7}^{a}(0)P_{0}. \tag{34}$$

All the joints of the robot are one freedom; so, the setting error should meet: $\|w_{i} + \delta w_{i}{}'\| = 1$, $\left(w_{i} + \delta w_{i}{}'\right)^{T}\left(v_{i} + \delta v_{i}{}'\right) = 0$, $s_{i} = (v_{i}, w_{i})^{T}$, and $\delta s_{i}{}' = (\delta v_{i}{}', \delta w_{i}{}')^{T}$, as shown in Table 2.

(a) The position error in the x direction

(b) The position error in the y direction

(c) The position error in the z direction
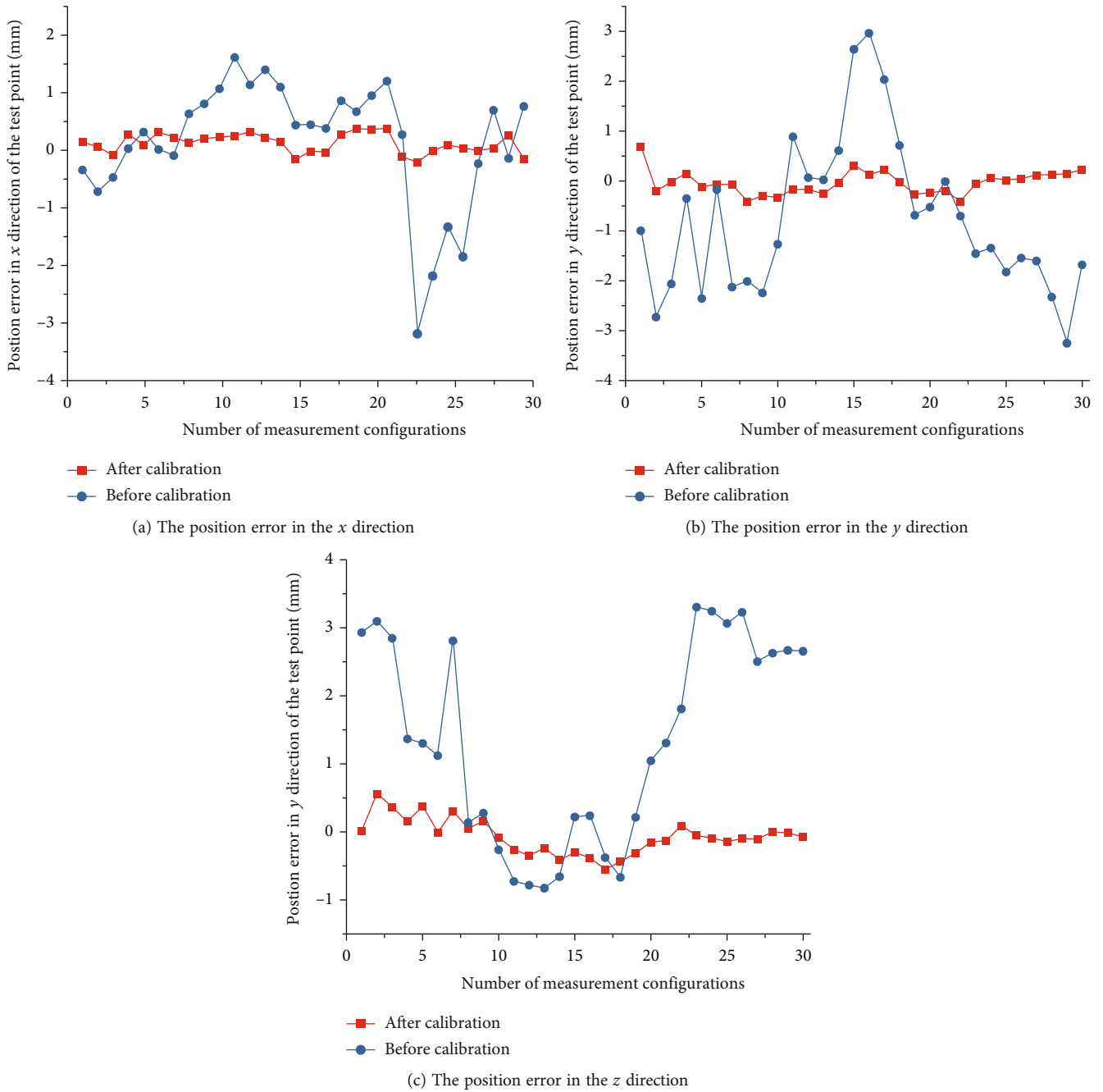
FIGURE 9: The position error in directions of x, y, and z before and after calibration in experiments.

100 groups of joint angle data are randomly generated, which are substituted to (34) to obtain the actual position data. The position data are divided into two groups: identification group and verification group. With the identification group of position data, the process of iterative convergence is shown in Figure 6 which shows that after 1st iteration, the objective function is reduced greatly; after 10 iterations, the maximum error is reduced from 29.597 mm to 0.146 mm, and the average error is reduced from 6.831 mm to 0.037 mm. The identified parameters are substituted to the kinematic model of the robot to calculate the position error in the three directions of the coordinate system with

the verification group of data, and the results are shown in Figure 7 which shows that the maximum error in x, y, and z directions is reduced from 19.797 mm to 0.094 mm, 26.548 mm to 0.092 mm, and 14.626 mm to 0.095 mm, respectively.

5. Experiments

To further verify the calibration method based on position information and LPoE model proposed in this paper, a calibration experiment was carried out on a six degree-of-freedom collaborative robot, the repeated positioning

accuracy of which is 0.03 mm. The measuring instrument is RADIAN Pro R20 of API, and the measuring radium and accuracy are 20 m and 15 μm + 0.7 μm/m, respectively. The measuring software is Spatial Analyzer (SA) of New River Kinematics. The experimental system is shown in Figure 8.

100 groups of joint angles are generated in the measurable space of the laser tracker, 70 groups of which are identification sets and the other 30 groups are verification sets. According to Figure 5, the kinematic parameters are identified, and the verification results are shown in Figure 9. The results show that after calibration, the maximum error in x, y, and z directions is reduced from 3.171 mm to 0.393 mm, 3.258 mm to 0.678 mm, and 3.297 mm to 0.560 mm, respectively. Therefore, the accuracy of the collaborative robot is increased greatly by the proposed calibration method based on position data and improved LPoE.

6. Conclusions

The kinematic model of the collaborative robot based on the LPoE model was established, and the corresponding error model and calibration method were derived. The calibration method was verified by simulation, which proved the validity of the error model and the calibration method. A six degree-of-freedom collaborative robot was used as the experimental objective, and the API RADIAN Pro R20 laser tracker was used as the measurement equipment to perform kinematic parameter calibration experiments which show that the positioning errors of the six degree-of-freedom collaborative robot decrease from 3.171 mm, 3.258 mm, and 3.297 mm to 0.393 mm, 0.678 mm, and 0.560 mm in the directions of x, y, and z, respectively, after kinematic parameter calibration. Therefore, the proposed calibration method is effective.

However, the residuals are still about 0.6 mm after calibration. The main reason is that kinematic calibration cannot eliminate the influence of nongeometric errors such as the elastic deformation of the joint. In the future, relevant research work will be carried out on nongeometric errors to further improve the positioning accuracy of the robot.

Data Availability

Data were curated by the authors and are available upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] ISO, *TS 15066: 2016: Robots and robotic devices–Collaborative robots*, International Organization for Standardization, Geneva, Switzerland, 2016.

[2] L. Fu, R. Wu, and J. Zhao, "The evolution and enlightenment of safety specification of cooperative robots: ISO/TS 15066," *Robot*, vol. 39, no. 4, pp. 532–540, 2017.

[3] Q. Liu, A. Liu, W. Meng, Q. Ai, and S. Q. Xie, "Hierarchical compliance control of a soft ankle rehabilitation robot actuated by pneumatic muscles," *Frontiers in Neurorobotics*, vol. 11, p. 64, 2017.

[4] W. Liu, D. Chen, and L. Zhang, "Trajectory generation and adjustment method for robot manipulators in human-robot collaboration," *Robot*, vol. 38, no. 4, pp. 504–512, 2016.

[5] X. Liu, F. Xie, and J. Wang, "Current opportunities in the field of mechanisms in China," *Journal of Mechanical Engineering*, vol. 51, no. 13, pp. 2–12, 2015.

[6] L. Li, Z. Luo, F. X. He, K. Sun, and X. L. Yan, "An improved partial similitude method for dynamic characteristic of rotor systems based on Levenberg-Marquardt method," *Mechanical Systems and Signal Processing*, vol. 165, article 108405, 2022.

[7] X. Liu, Q. Liu, C. Yue et al., "Intelligent machining technology in cutting process," *Journal of Mechanical Engineering*, vol. 54, no. 16, pp. 45–61, 2018.

[8] Z. Fu, J. S. Dai, K. Yang, X. Chen, and P. López-Custodio, "Analysis of unified error model and simulated parameters calibration for robotic machining based on Lie theory," *Robotics and Computer-Integrated Manufacturing*, vol. 61, article 101855, 2020.

[9] W. Gao, H. Wang, Y. Jiang, and X. Pan, "Research on the calibration for a modular robot," *Journal of Mechanical Engineering*, vol. 50, no. 3, pp. 33–40, 2014.

[10] B. Tipary and G. Erdős, "Generic development methodology for flexible robotic pick-and-place workcells based on Digital Twin," *Robotics and Computer-Integrated Manufacturing*, vol. 71, article 102140, 2021.

[11] Z. Xie, S. Xin, X. Li, and S. Xu, "Method of robot calibration based on monocular vision," *Journal of Mechanical Engineering*, vol. 47, no. 5, pp. 35–39, 2011.

[12] Z. Jiang, W. Zhou, H. Li, Y. Mo, W. Ni, and Q. Huang, "A new kind of accurate calibration method for robotic kinematic parameters based on the extended Kalman and particle filter algorithm," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3337–3345, 2017.

[13] W. Li, "Analysis on Kinematic Model of Welding Robot Calibration for Automobile Fittings," in *8th International Conference on Applied Science, Engineering and Technology (ICASET)*, pp. 146–150, Qingdao, China, 2018.

[14] Y. Lu, L. Yu, and B. Guo, "Calibration of industrial robot structure parameters based on closed dimensional chain," *Chinese Journal of Scientific Instrument*, vol. 39, no. 2, pp. 38–46, 2018.

[15] K. Okamura and F. Park, "Kinematic calibration using the product of exponentials formula," *Robotica*, vol. 14, no. 4, pp. 415–421, 1996.

[16] R. He, Y. Zhao, F. Han, S. Yang, and S. Yang, "Experimentation on identifying the kinematic parameters of serial mechanism based on the product-of-exponential formula," *Robot*, vol. 33, no. 1, p. 35, 2011.

[17] R. He, Y. Zhao, S. Yang, and S. Yang, "Kinematic-parameter identification for serial-robot calibration based on POE

formula," *IEEE Transactions on Robotics*, vol. 26, no. 3, pp. 411–423, 2010.

[18] Y. Jiang, L. Yu, Y. Chang, H. Jia, and H. Zhao, "Robot calibration based on modified differential evolution algorithm," *Optics and Precision Engineering*, vol. 29, no. 7, pp. 1580–1588, 2021.

[19] Y. Wang, Q. Wei, C. Hu, and X. Ding, "A self-calibration method of space manipulators based on POE formula," *Journal of Beijing University of Aeronautics and Astronautics*, vol. 44, no. 11, pp. 2336–2342, 2018.

[20] G. Chen, Q. Jia, T. Li, and H. Sun, "Calibration method and experiments of robot kinematics parameters based on error model," *Robot*, vol. 34, no. 6, pp. 680–688, 2012.

[21] I. Chen, G. Yang, C. T. Tan, and S. H. Yeo, "Local POE model for robot kinematic calibration," *Mechanism and Machine Theory*, vol. 36, no. 11-12, pp. 1215–1239, 2001.

[22] R. M. Murray, Z. Li, and S. S. Sastry, *A mathematical introduction to robotic manipulation*, CRC, Boca Raton, FL, USA, 1994.

Research Article

A Novel Trajectory Tracking Control with Modified Supertwisting Sliding Mode for Human-Robot Cooperation Manipulator in Assembly Line

Xingyu Wang [ID],[1] Anna Wang [ID],[1] Dazhi Wang,[1] Zhen Liu,[1] and Yufei Qi[2]

[1]*College of Information Science and Engineering, Northeastern University, Shenyang 110819, China*
[2]*China North Vehicle Research Institute, Beijing 100072, China*

Correspondence should be addressed to Anna Wang; wanganna@mail.neu.edu.cn

In this paper, the trajectory tracking problem of industrial manipulator with fixed periodic external force on human-robot cooperative assembly line is studied. An advanced fuzzy adaptive supertwisting sliding mode control (FASTSMC) algorithm is proposed to realize the rapid convergence and continuous control of nonlinear control system. In order to enhance the robustness of the system, an arctangent terminal sliding mode surface is designed to deal with the concentrated uncertain disturbance. The supertwisting method is used to overcome the chattering problem in the control law. The fuzzy adaptive technique is used to compensate the centralized disturbance with unknown upper bound. A stiffness identification model is designed to estimate the deviation of the manipulator under external force. Finally, the feasibility of the proposed control scheme is verified by a simulation example of a 4-DOF manipulator.

1. Introduction

The development of Industry 4.0 has led to great changes in manufacturing specifications and aims to reduce human participation by replacing industrial robots to perform the most repetitive tasks [1]. Human-robot cooperation is a representative technology, which has become the key technology of the future factory. For example, this technology has been used in assembly lines in the automotive industry such as the Volkswagen factory in Wolfsburg, Germany. It combines the advantages of human workers and assistant robots and allows different degrees of automation in the workplace to meet the increasing flexibility needs of manufacturing systems. In the hybrid team of human and robot, the organization needs intelligent planning and control algorithms. The system must be reconfigured quickly to meet the needs of high production of different products, but human participation is inevitable [2]. Human-robot cooperative system is a nonlinear time-varying dynamic system with uncertain interference, in which the uncertainty and external interference greatly improve the complexity of the system. Many

researchers have designed corresponding controllers for this technical problem, such as variable impedance control method [3], adaptive admittance controller [4], underactuated redundant low impedance method [5], and efficiency weighting strategy [6]. However, the existing methods lack ideal solutions to three main challenges: (1) the control system cannot guarantee the convergence of stable equilibrium. (2) The robustness of human-robot cooperative system is endangered by uncertainty and external interference. (3) The nature and characteristics of external interference are difficult to identify and compensate.

Although many advanced control methods, such as adaptive control [7], neural network control [8], and variable structure control [9], can be applied to human-robot cooperation manipulator control system, they all have some limitations. Deng [10] developed a novel finite-time command filter backstepping method, which allows the traditional command filter backstepping control and ensures the finite-time convergence. Figueredo et al. [11] proposed a robust dual-quaternion-based H-infinity task-space kinematic controller for robot manipulators, and this controller

provides higher precision and faster response than the conventional H-infinity control. Although the above schemes have achieved some positive conclusions, the design of these controllers is complex, and the steps of adjusting parameters in practical engineering application are cumbersome, which is not conducive to wide popularization. Therefore, this paper attempts to improve the more easily realized sliding mode control and apply it to the end trajectory tracking control of human-robot cooperation manipulator.

The conventional sliding mode control principle is to use the symbolic function to make a decision on the input signal and then process the feedback, respectively, so that the control system can obtain strong robustness. The conventional linear sliding mode control can only achieve the asymptotic convergence of the state. Generally, increasing the gain to make the actuator saturate quickly is a common method to accelerate the convergence speed of tracking error [12]. In order to make the sliding mode control more in line with the needs of practical application, many improved schemes of sliding mode control have been proposed one after another. In reference [13], a terminal sliding mode control of PID type is proposed to strengthen the nonlinear relationship between state variables and control inputs. In reference [14], combining the sliding mode control method and the fast exponential convergence method, the sliding mode surface can be obtained faster, and the tracking error converges in the form of fast exponential convergence. In addition, terminal sliding mode control has the characteristics of reduced control gain, strong anti-interference ability, and finite-time convergence. It is also often used in the control of manipulator. For example, based on the terminal sliding mode control scheme proposed in reference [15], the low gain control of the manipulator system is realized. For the stabilization of underactuated robotic systems, reference [16] designed a fast terminal sliding mode control technology based on disturbance observer, estimated the external disturbance of the system by using the finite-time disturbance observer, and established the finite-time control law. In order to eliminate the singularity in the control process, a nonsingular continuous terminal sliding mode control system is designed by using the power reaching law in reference [17]. Tracking control of manipulator system can be realized by establishing reference signal and synchronizing it. Reference [18] uses neural network technology and sliding mode control method to design controller. In reference [19], the combination of sliding mode control method and fault-tolerant control is adopted. In the design of the manipulator controller, the supertwisting method is used to eliminate the control chattering and realize the finite-time tracking control. An adaptive finite-time sliding mode control is applied to chameleon chaotic systems with uncertainties and disturbances [20]. Alattas et al. [21] applied the optimal integral terminal sliding mode control based on the control Lyapunov method to the asymmetric nonholonomic robot system with external disturbances and obtained positive conclusions. In reference [22], the finite-time tracking control is realized by formulating the boundary layer. The sliding mode control method realizes quasi-sliding mode control by adjusting symbolic function and exponential function. However, this method

has the disadvantage of reducing the dynamic accuracy of the system.

There are many research achievements in the end trajectory tracking control of manipulator based on sliding mode control scheme and the solution of human-robot cooperation. In reference [23], an impedance control structure for monitoring the contact force between the end effector and the environment is proposed, and the model-free fuzzy sliding mode control strategy is used to design the position controller and force controller. A fuzzy controller is proposed in reference [24]. A nonlinear model is used to track the trajectory of the micro robot in the human vascular system. In addition, the application of sliding mode control in the field of external force control of manipulator not only improves the robustness of the controller but also improves the estimation of force through sliding disturbance observer, so as to avoid the use of expensive force sensors in connecting rods and joints, such as reference [25]. The estimation of force is closely related to the stiffness identification of manipulator link and joint. Some studies identify the change of manipulator stiffness matrix and then calculate the change of external force, but it requires high accuracy of sensor for stiffness measurement [26, 27]. In the above research, in addition to the research on the manipulator control system in the fixed application field, the impact of stiffness change on the end trajectory tracking control effect is rarely considered in the related research using the general manipulator dynamic model. The main difficulty is that special sensors need to be added to measure the change of stiffness, and the installation position of the sensor is difficult to determine because the trigger point of the change of manipulator link or joint stiffness is not fixed in different application environments. On the premise of considering the elastic deformation law under the influence of external force, the elastic deformation law model is introduced into the general manipulator model, and the compensation mechanism is introduced. The most common case of periodic variation of elastic deformation is the scene where the manipulator works repeatedly on the assembly line.

During the observation of the production process, it is found that in the common scene of human-robot cooperation in industrial production, the external force generated by man on the manipulator often has a periodic law, which often occurs in the assembly line. Because the operation of the workpiece in the assembly line is cyclic and unified, the physical contact rate between man and the manipulator will appear in a specific action. Therefore, this paper believes that the stiffness change caused by this external force can be summarized into the corresponding model. The main innovation of this paper is to solve the elastic deformation law of manipulator joints and links, combined with the stiffness identification model, and combined with the general dynamic model of manipulator. The trajectory deviation caused by external force contact is no longer regarded as uncertain disturbance, which improves the performance of the controller. In addition, the fuzzy adaptive supertwisting sliding mode controller proposed in this paper uses the arctangent function to replace the symbolic function, which improves the performance of the controller. In addition, 2-DOF and 4-

DOF manipulators are used to verify the effectiveness of the proposed algorithm and control scheme. From the perspective of practical application, the scheme proposed in this paper does not need to add additional sensors on the manipulator and will not increase the economic cost of industrial production.

2. Model Derivation and Control Objectives

Based on the general dynamic equation of the manipulator, the dynamic equation of the n-DOF manipulator under external disturbance can be obtained as follows [28].

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) = \tau + \tau_d, \tag{1}$$

where $q, \dot{q},$ and $\ddot{q} \in R^n$ are the vectors of joint positions, velocities, and accelerations, respectively. $M(q) \in R^{n \times n}$ is the positive definite inertia matrix of the manipulator; $C(q,\dot{q}) \in R^{n \times n}$ is the Coriolis force and centrifugal force matrix of the manipulator; $G(q) \in R^n$ is the gravity matrix of the manipulator; $F(\dot{q}) \in R^n$ is the friction matrix; $\tau \in R^n$ is the control torque of the manipulator joint; $\tau_d \in R^n$ is the uncertain external disturbances. In practical industrial applications, the values of these physical parameters cannot be accurately obtained due to modelling errors, payload changes, external interference, other inherent factors, and human factors. Therefore, the uncertainty caused by the system parameter error can be expressed as

$$\begin{cases} M(q) = M_0(q) - \Delta M(q), \\ C(q,\dot{q}) = C_0(q,\dot{q}) - \Delta C(q,\dot{q}), \\ G(q) = G_0(q) - \Delta G(q), \\ F(\dot{q}) = F_0(\dot{q}) - \Delta F(\dot{q}), \end{cases} \tag{2}$$

where the subscript has the symbol of 0, such as M_0, which means the nominal matrix of manipulator parameters, and the symbol with Δ in front means the uncertain part of the robot arm control system, such as ΔM. Combined with formulas (1) and (2), a new dynamic equation of manipulator can be obtained.

$$\begin{aligned} M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) \\ = \tau + \tau_d + \Delta M(q)\ddot{q} + \Delta C(q,\dot{q})\dot{q} + \Delta G(q) + \Delta F(\dot{q}). \end{aligned} \tag{3}$$

Then, through the above analysis, it can be concluded that in the presence of uncertainty and interference, the dynamic equation (3) can be expressed as

$$\begin{aligned} \ddot{q} = M_0(q)^{-1}\{-C_0(q,\dot{q})\dot{q} - G_0(q) - F_0(\dot{q}) + \tau + \tau_d \\ + \Delta M(q)\ddot{q} + \Delta C(q,\dot{q})\dot{q} + \Delta G(q) + \Delta F(\dot{q})\}. \end{aligned} \tag{4}$$

In order to facilitate representation and calculation, a simplified dynamic model can be obtained according to equation (4), which is written as

$$\ddot{q} = f_0(q,\dot{q}) + M_0(q)^{-1}\tau(t) + D(t), \tag{5}$$

where $D(t)$ is the uncertainty and disturbance in the manipulator control system and $f_0(q,\dot{q})$ represents a nonlinear function with known boundary.

As can be seen from (1), the parameters in the manipulator control system are highly coupled and are also affected by dynamic time-varying parameters, which will affect the stability of the system. In practical engineering, it is very difficult to accurately obtain the mathematical model, such as unknown parameters, unmodeled nonlinear functions, and payload changes. However, some prior knowledge of the model can be obtained by analyzing the dynamic time-varying characteristics. Therefore, in the design of model-based manipulator controller, we should not give up considering the influence of unknown disturbances but summarize the corresponding approximate mathematical model through the existing data. Based on the above classical model, this paper considers the cooperative work of the manipulator in the task, and this cooperation is different from the parallel manipulator in most cases. It may be human-robot cooperation or the cooperative work of multiple independent mechanical systems, so it is inevitable to have direct or indirect mutual contact and collision. In this paper, the resulting external force and influence are expressed by the following equation.

$$F_e = M\ddot{\tilde{x}} + D\dot{\tilde{x}} + K\tilde{x}, \tag{6}$$

where F_e is the external generalized force exerted by the manipulator on the environment and $K \in R^{n \times n}$ is the stiffness matrix. While $\tilde{x} = x_r - x$ is the end effector tracking error with x_r being the reference trajectory, it is worth noting that there is a difference in meaning between the end trajectory error here and the end position error later. The trajectory error here refers to the trajectory error generated during the execution of the task, and the end position error later refers to the signal error between the controller and the sensor.

For the end control of the manipulator, the goal is to design a robust sliding mode control to ensure high-precision tracking of the desired known trajectory of the manipulator in the presence of uncertainty and external interference.

$$\lim_{t \to t_f} |e| = \lim_{t \to t_f} |q - q_d| = 0, \tag{7}$$

where $q_d \in R^n$ is a desired trajectory, $e = q - q_d$ is trajectory tracking error, and t_f is finite time. Then, the controller is designed according to the control objectives mentioned in the paper. In order to ensure the rationality of the designed controller, the stability is analyzed through the following basic assumptions and lemmas.

The assumptions are as follows:

(i) Joint position and speed status are data that can be measured by manipulator sensors

(ii) The total uncertainty $\psi D(t)$ satisfies the following conditions, $|\psi D(t)| \le \omega_0$ and $|\psi \dot{D}(t)| \le \omega_1$, where ψ

is a constant diagonal matrix, which is formulated in advance. Constants ω_0 and ω_1 are unknown and positive

Through the above assumptions and according to the relevant lemmas, the stability of the following invariant system can be proved.

$$\dot{z} = f(z). \tag{8}$$

Suppose there is a continuous function $V(z)$ in the system shown in equation (7), which satisfies the following conditions, i.e., it is a positive definite function on $D \subseteq R^n$ and satisfies constraint condition:

$$\dot{V}(z) + kV^{\lambda}(z) \le 0, \tag{9}$$

where $k > 0$ and $\lambda \in (0, 1)$, it can be concluded that the system based on equation (7) is locally finite time stable. If $D = R^n$ is satisfied, the system is globally finite time stable.

3. FASTSMC of Manipulator with External Force Prediction

3.1. Design of Sliding Surface.
The nonlinear sliding surface describing the manipulator control system in equation (5) can be defined as

$$\sigma(t) = \psi\left(\theta(t) - \theta(0)e^{-\beta t}\right), \tag{10}$$

$$\theta(t) = \dot{e}(t) + \lambda e(t), \tag{11}$$

where $e(t) = q - q_d$ is the joint space tracking position error with $q_d \in R^n$ as the known desired position trajectory. From these, we can get the result that $\dot{e}(t) = \dot{q} - \dot{q}_d$ represents the tracking speed error of the joint, $\dot{q}_d$ means the desired speed trajectory given by the control system, ψ is a constant diagonal matrix, and λ and β are the positive coefficients. Then, the derivative of equation (10) with respect to t can be obtained.

$$\dot{\sigma}(t) = \psi\left(\dot{\theta}(t) + \beta\theta(0)e^{-\beta t}\right) = \psi\left(\ddot{e}(t) + \lambda\dot{e}(t) + \beta\theta(0)e^{-\beta t}\right). \tag{12}$$

Combining equation (5), equation (12) can be rewritten as

$$\dot{\sigma}(t) = \psi\left(f_0(q,\dot{q}) - \ddot{q}_d + \lambda\dot{e}(t) + \beta\theta(0)e^{-\beta t} + M_0(q)^{-1}\tau(t) + D(t)\right). \tag{13}$$

The above formula can be further simplified to obtain

$$\dot{\sigma}(t) = \psi\left(f_0(q,\dot{q}) - \ddot{q}_d + \lambda\dot{e}(t) + \beta\theta(0)e^{-\beta t}\right) \\ + \psi M_0(q)^{-1} + \psi D(t) = f(t) + h(t)\tau(t) + \omega(t). \tag{14}$$

3.2. Sliding Mode Controller.
Firstly, the conventional supertwisting algorithm formula is rewritten as

$$\dot{\sigma}(t) = -l_1(t)\Lambda(\sigma(t))\,\text{sgn}\,(\sigma(t)) + \varpi(t), \tag{15}$$

$$\dot{\varpi}(t) = -l_2(t)\,\text{sgn}\,(\sigma(t)) + \eta(t), \tag{16}$$

where $\Lambda(\sigma(t)) = \text{diag}\,[|\sigma_1(t)|^{1/2} \cdots |\sigma_n(t)|^{1/2}]$, $l_1 = \text{diag}\,[l_{11} \cdots l_{1n}]$, and $l_2 = \text{diag}[l_{21} \cdots l_{2n}]$ are diagonal positive matrices and $\eta(t)$ is the unknown bounded uncertainties, same as $\psi D(t)$ stated in the above assumptions. Moreover, it is necessary to keep $\sigma(t) = \dot{\sigma}(t) = 0$ for a finite time [29].

When the uncertainty boundary ω_0 of the error is known, the parameters l_1 and l_2 can be taken according to the method of reference [17], i.e., $l_1 = \text{diag}\,[1.5\sqrt{\omega_0} \cdots 1.5\sqrt{\omega_0}]$ and $l_2 = \text{diag}\,[1.1\omega_0 \cdots 1.1\omega_0]$. It should be added that in the above case, l_1 and l_2 are constants, and the gain l_2 must be greater than the boundary ω_0 in order to switch the term $l_2\,\text{sgn}\,(\sigma(t))$. The uncertainty term $\eta(t)$ can be controlled. It is worth noting that in many cases, the control gain may be overestimated, resulting in increased chattering. Later, we will design a fuzzy adaptive compensation method to reduce chattering.

According to the conclusion in reference [18], equivalent control on the sliding surface means average control. Once the control system reaches the sliding surface, equivalent control will occur. In order to keep the system moving on the sliding surface, the sliding mode gain should not be less than the upper limit of uncertainty. Obviously, due to the switching characteristics of sliding mode control, the motion of the system on the sliding model surface is discontinuous, but the system controller needs to be continuous and equivalent approximate. Therefore, it is necessary to use a low-pass filter to obtain equivalent control.

$$s(t) = K(t) - \frac{1}{\mu\beta_0}|u_{eq}(t)| - \epsilon, \tag{17}$$

where $0 < \mu < 1/\beta_0 < 1$ and $\epsilon > 0$ A is a positive scalar and cannot be too large and $u_{eq}(t) = \eta(t) = l_2\,\text{sgn}\,(\sigma(t))$ is equivalent control on the sliding surface. The proposed adaptive control law $K(t)$ is defined as

$$K(t) = k_0 + \dot{k}(t), \tag{18}$$

$$k(t) = -\varrho(t)\,\text{sgn}\,(s(t)), \tag{19}$$

$$\varrho(t) = r_0 + r(t), \tag{20}$$

$$r(t) = \gamma|s(t)|, \tag{21}$$

where k_0, r_0, and γ are positive constants and $\varrho(t)$ is a time-varying scalar.

In (15) and (16), discontinuous symbolic functions exacerbate the chattering of sliding mode control. For sliding mode control, the effective method to suppress chattering is to select the appropriate switching function to improve its continuity and smoothness. The common method is to replace the traditional symbolic function with the saturation

function, so that the boundary layer can be a continuous state linear feedback control, but there is a problem of uneven transition stage, resulting in drastic changes in slope. In order to improve the smoothness of boundary layer transition stage, some recent studies use hyperbolic tangent function as switching function. As shown in the figure, the transition of hyperbolic tangent function is smooth in the boundary layer, and the curve slope in the transition stage of hyperbolic tangent function changes little near the boundary layer. In addition, it is a continuous state feedback control in the boundary layer to improve the smoothness.

Different from the existing sliding mode control, this paper uses the arctangent function as the switching function. It can be seen from Figure 1 that the characteristics of the arctangent function and the hyperbolic tangent function are similar when they are used as the switching function. However, there is an exponential function in the hyperbolic tangent function. In the operation process, especially when calculating the derivative, the amount of operation is large, and the occupation of operation resources is more obvious when calculating the high-dimensional model. Therefore, considering the above reasons, equation (19) in system (17) is rewritten as

$$\dot{k}(t) = -\varrho(t) \arctan(s(t)). \tag{22}$$

Considering that the robotic manipulator system (1) satisfies the above assumptions and proposes the control equation as equation (17), it can guarantee the finite-time convergence of the error $e(t)$ to zero point, and it can track even in the presence of uncertainty and external disturbances asymptotically expected known trajectory q_d.

We can get the convergence of systems (15) and (16) in a finite time. Especially $\sigma(t) = 0$, then from equation (10), we can get the following result:

$$\theta(t) = \theta(t)(0) \exp(-\beta t). \tag{23}$$

Then, (23) can be regarded as the solution of the following first-order differential equation:

$$\dot{\theta}(t) + \beta\theta(t) = 0. \tag{24}$$

Since $\beta > 0$, $\theta(t)$ will converge gradually to zero; therefore, it can be inferred that when $t \longrightarrow \infty$, $\theta(t) \longrightarrow 0$, and this will cause $e(t) \longrightarrow 0$ and $q \longrightarrow q_d$. This just verifies the closed-loop stability of the controller in the manipulator system.

3.3. Fuzzy Adaptive Sliding Mode Controller. Sliding mode control is an effective control technology to solve the zero dynamic problem of nonminimum phase nonlinear systems. However, in these designs, the concern about chattering has not been completely solved without reducing the robustness of the closed-loop system. High-order sliding mode control and integral sliding mode control are alternative control methods to effectively solve the chattering problem and further enhance the robustness of the closed-loop system. However, the design of these controllers still requires a priori
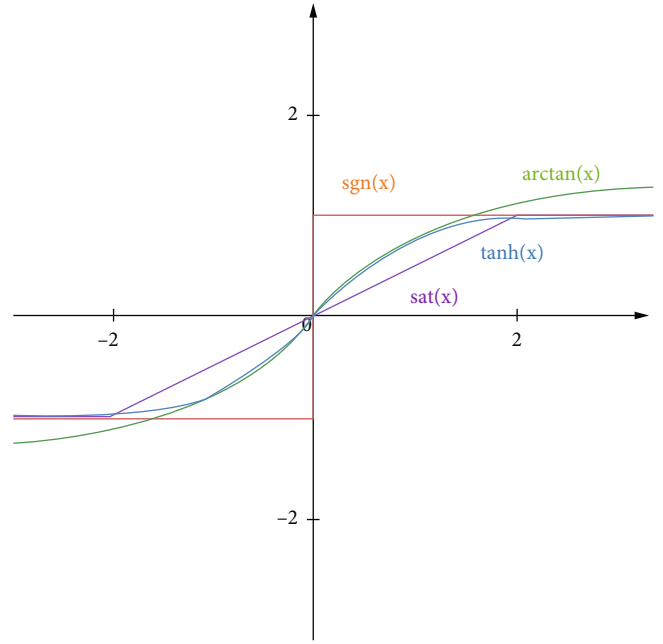
FIGURE 1: Switching function.

knowledge of the uncertainty boundary. This paper adopts the construction of fuzzy rules as the formulation standard of uncertainty boundary in controller design, and these rules and membership functions often rely on empirical knowledge.

Aiming at the problem of manipulator control with unknown model uncertainty and external interference, this paper uses fuzzy logic inference algorithm to compensate for the unknown uncertainty of the system. Among them, the fuzzy function contains the adaptive parameters of sliding mode control to realize the adaptive sliding mode control.

The sliding surface $\theta(t)$ and $\dot{\theta}(t)$ are selected as the inputs of the fuzzy system. Input and output fuzzy sets are defined as $\{NL \quad NM \quad ZO \quad PM \quad PL\}$, where NL represents negative large, NM represents negative medium, ZO represents zero, PM represents positive medium, and PL represents positive large. The Gaussian membership function of fuzzy system can be expressed as

$$\mu_z(\Omega_i) = \exp\left[-\left(\frac{\Omega_i - \alpha}{\rho}\right)^2\right], \tag{25}$$

where Ω_i is the input and output of the fuzzy system, the subscript z of the membership function represents the fuzzy rule set, α and ρ represent the center and width of the fuzzy rule set z, respectively.

When the state trajectory deviates from the sliding mode surface or the approaching speed is small, the output should be increased appropriately to make it reach the sliding mode surface quickly. When the state trajectory approaches the sliding surface or the approaching speed is large, the output should be appropriately reduced to suppress chattering.

According to the above analysis, the determination of fuzzy rules is shown in Table 1.

In the rule base, the central average defuzzification and product reasoning methods are adopted, and the output of the fuzzy system can be expressed as

$$y = \frac{\sum_{j=1}^{m} (\theta_j) \left(\prod_{i=1}^{n} \mu_i(x_i) \right)}{\sum_{j=1}^{m} \left(\prod_{i=1}^{n} \mu_i(x_i) \right)} = \Gamma^T \Psi, \qquad (26)$$

where $\Gamma = \begin{bmatrix} \Gamma_1 & \cdots & \Gamma_i & \cdots & \Gamma_m \end{bmatrix}^T$ and $\Psi = \begin{bmatrix} \Psi_1 & \cdots & \Psi_i & \cdots & \Psi_m \end{bmatrix}^T$.

This paper defined Γ_i shown as

$$\Gamma_i(x) = \frac{\prod_{i=1}^{n} \mu_i(x_i)}{\sum_{j=1}^{n} \left(\prod_{i=1}^{n} \mu_i(x_i) \right)}. \qquad (27)$$

Therefore, the fuzzy function can be obtained.

$$P = \Gamma^T(x) \Psi, \qquad (28)$$

where Γ^T is a membership function and Ψ is an adaptive fuzzy parameter. The adaptive fuzzy parameters are formulated as follows:

$$\dot{\Psi} = \delta \left(M^{-1}(q) \right)^T \Gamma_s, \qquad (29)$$

where $\delta > 0$ is a scalar design parameter.

3.4. Simulation. This part will take a 2-link manipulator as an example to complete a simple trajectory, observe the effectiveness of the proposed FASTSMC in the basic example, and compare it with the classical sliding mode control [30] and the conventional supertwisting sliding mode control [31] to reflect the application performance of the proposed method. The desired trajectory given by this simulation is $q_d = \begin{bmatrix} \cos(0.5t) & 2\sin(0.5t) \end{bmatrix}^T$.

It can be seen from Figure 2 that the tracking effect of classical sliding mode control is poor, the convergence speed is slow, and it is not stable enough in the tracking process. After stabilization, some errors will still occur sometimes. Supertwisting sliding mode control is a high-order sliding mode control method. It has the advantage of realizing the finite-time convergence of sliding mode variables and their derivatives, can avoid the singularity problem existing in other sliding mode control methods, and can make the chattering in the output control signal more subtle. As shown in Figure 3, the track tracking effect of supertwisting sliding mode control is very excellent, but the convergence speed is not satisfactory, which is difficult to apply in the case of uncertainty in the system. The FASTSMC proposed in this paper can quickly and accurately maintain trajectory tracking. Compared with the above two traditional methods, as shown in Figure 4, the method proposed in this paper has better performance in convergence speed and stability.

Then, the complexity of the algorithm proposed in this paper is shown in Table 2. The computational complexity of the algorithm proposed in this paper is expressed by the

TABLE 1: Fuzzy control rules.

s	$\dot{s}$				
	NB	NM	ZO	PM	PL
NL	PL	PL	PL	PM	ZO
NM	PL	PM	PM	ZO	NM
ZO	PM	PM	ZO	NM	NM
PM	PM	ZO	NM	NM	NM
PL	ZO	NM	NL	NL	NL

time required for operation in MATLAB. The simulation is run using MATLAB 2020a on a personal computer with Intel Core, CPU i7-7700 @ 3.60 GHz, and 8 GB RAM. It is not only compared with the above sliding mode control and supertwisting sliding mode control but also with the computational complexity of quantum interference artificial neural network proposed in [8]. In order to make the test results more accurate, each scheme was tested ten times, and the calculation time was taken as their average.

It can be seen from Table 2 that the processing time of the control scheme proposed in this paper is longer than that of some conventional sliding mode control schemes, which is mainly caused by MATLAB calling fuzzy logic module. Compared with other novel schemes, the method proposed in this paper has advantages in complexity.

3.5. Predict Joint Stiffness Model. In the manipulator control system with n rotating joints, the driving system of the joint can be regarded as a cantilever system, and its stiffness can be expressed by the spring constant k_θ. On this basis, the joint stiffness of the manipulator system can be expressed as the following diagonal matrix:

$$K = \text{diag} \left(k_{\theta_1}, k_{\theta_2}, \cdots, k_{\theta_n} \right). \qquad (30)$$

In the use of industrial manipulator, because the end effector of the manipulator is affected by external contact force, each joint will deform, resulting in the change of the direction of the end effector. By analyzing the mechanical characteristics of the industrial manipulator, it can be found that there is the following relationship between the elastic deformation at the joint of the manipulator and the torque applied at the joint:

$$\tau_k = K e_k(t), \qquad (31)$$

where $e_k(t)$ is the vector representing the deformation of joint angle under the influence of joint stiffness change.

Firstly, the stiffness of each joint of the manipulator is assumed to be known, and the model between each joint is established by Jacobian matrix, and the mapping relationship with the elastic deformation of the end joint is obtained. After that, through the kinematic model of the manipulator, the law between the end deformation of the manipulator and the deformation of other joints can be obtained as follows:

$$\Delta X = J e_k(t), \qquad (32)$$
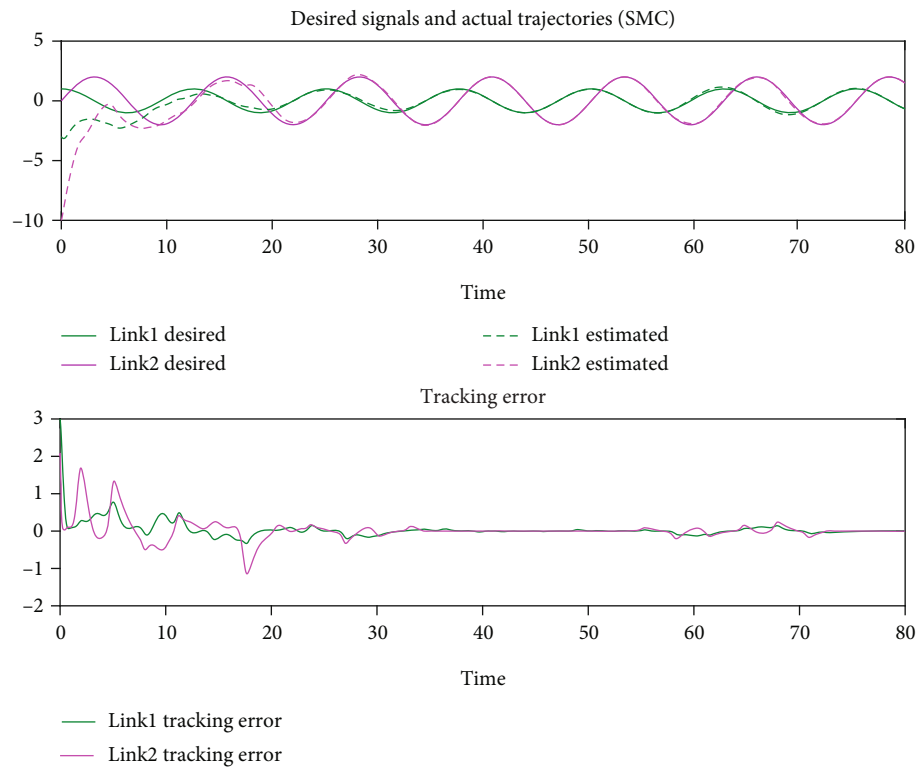
FIGURE 2: The joint trajectories and position errors of the manipulator with classical sliding mode control.
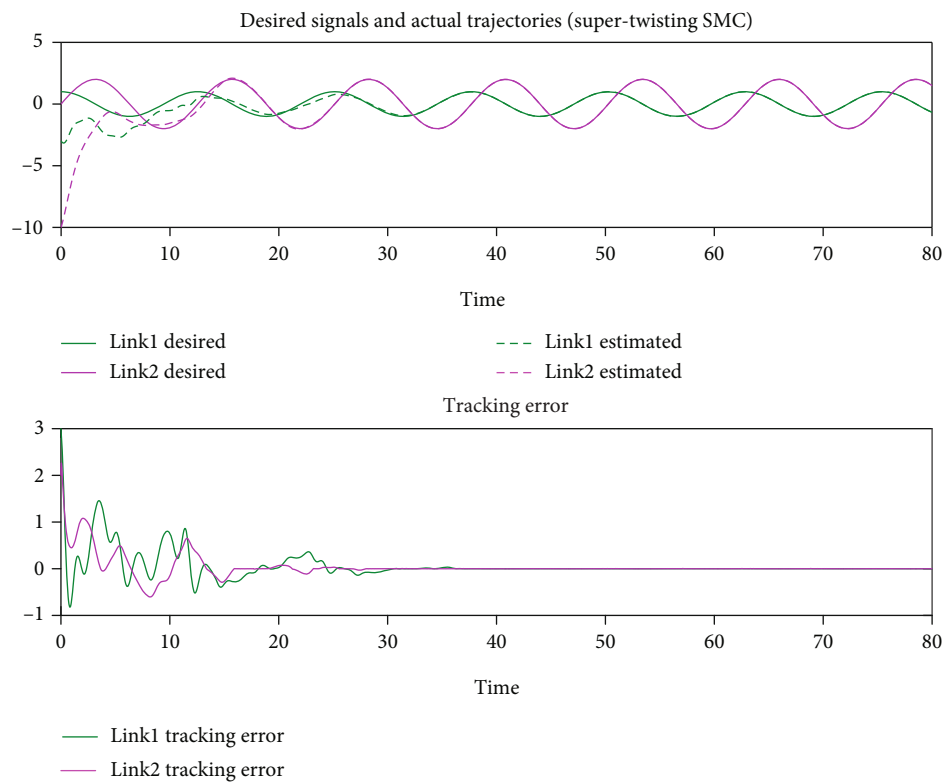
FIGURE 3: The joint trajectories and position errors of the manipulator with conventional supertwisting sliding mode control.

where ΔX is a vector used to describe the relationship between the joint deformation of the manipulator and the position deviation of the end actuator of the manipulator and J is the Jacobian matrix calculated by the vector product method according to the kinematic model of the manipulator.
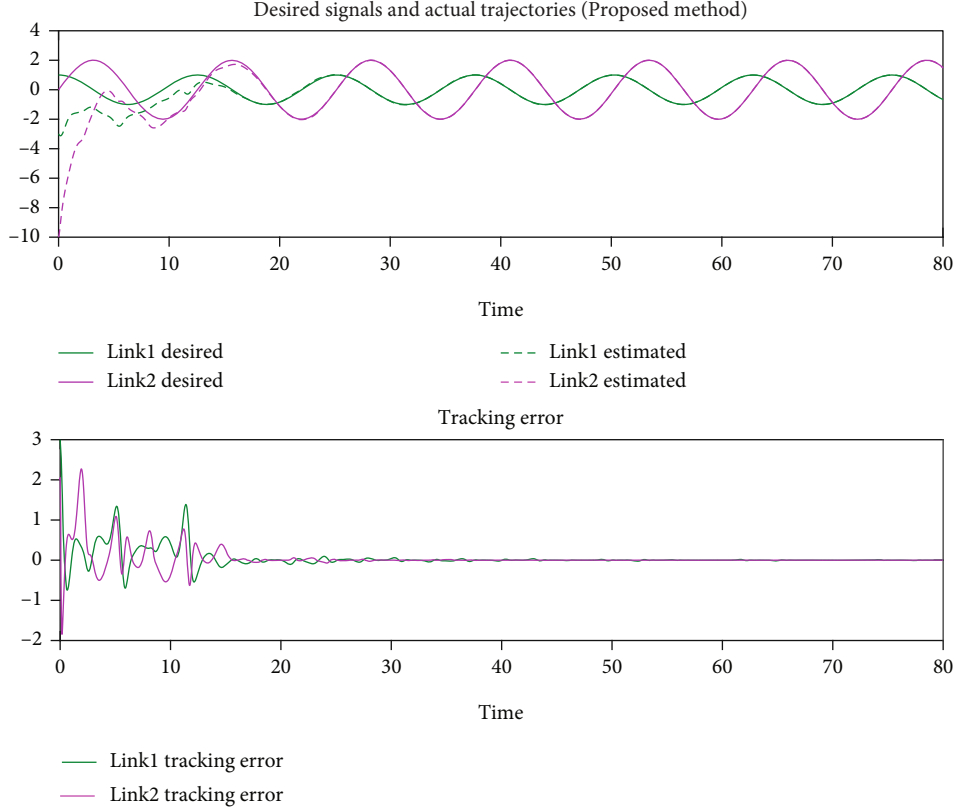
FIGURE 4: The joint trajectories and position errors of the manipulator with the method proposed in this paper.

TABLE 2: Comparison of processing time of control schemes.

Control scheme	Processing time (s)
Sliding mode control	2.230
Supertwisting sliding mode control	2.520
Quantum-interference artificial neural network	132.565
FASTSMC (proposed in this paper)	7.372

The influence of joint friction and gravity has been considered above, and these factors are ignored in this part. Therefore, the relationship between the joint torque caused by compensation deformation and the applied generalized force can be obtained:

$$\tau_k = J^T F_e. \tag{33}$$

Substituting (31) and (32) into (33), respectively, we can get

$$\Delta X = JK^{-1}J^T F_e. \tag{34}$$

Assuming that the Jacobian matrix and the terminal generalized force vector are known, we can obtain the joint deformation mapping of the manipulator in the workspace and calculate the joint torque caused by the change of joint stiffness caused by external force. Therefore, we define a

model predictive control, which links the deformation law of the rigid body with the sliding mode control above. The system assumption of the stiffness part has no uncertainty, and the dynamics is given by a simple integrator chain and controlled by v.

$$v = \underset{}{\arg\min} \sum_{j=0}^{d} \left\| \dot{\sigma}_{k+j} \right\|^2. \tag{35}$$

If there is no activation constraint, the nominal solution can be obtained from the model predictive control conclusion v by minimizing the square norm of the prediction derivative of the sliding vector evaluated using the nominal dynamics. Otherwise, a compromise will be reached to meet the constraints. The prediction derivative can be easily calculated from the state prediction without uncertainty [32].

$$\dot{\sigma}_k = \begin{bmatrix} K & D & M \end{bmatrix} \left(\begin{bmatrix} x_r \\ \dot{x}_r \\ \ddot{x}_r \end{bmatrix} - \begin{bmatrix} x \\ J\dot{q} \\ J\dot{q} + Jv \end{bmatrix} \right) - F_e. \tag{36}$$

The predicted values referenced by the model are direct because they are designed by the user. For external forces without high-order information, it is obvious that the longer the delay time, the less effective the assumption is, resulting in performance degradation during contact transients. The
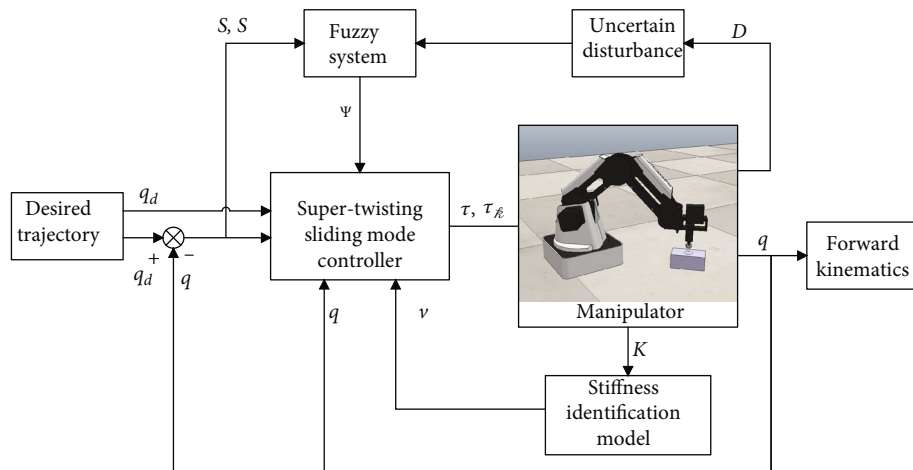
FIGURE 5: The structure diagram of the controller.

TABLE 3: Parameters used in controller simulation.

Type	Parameter	Value
Controller parameters	λ (in equation (11))	5
	$\beta_0, k_0, r_0, \gamma$ (in equations (17)-(21))	1.99, 3, 3, and 5
	ω_0 (in equations (15) and (16))	4
	α, ρ (in equation (25))	3, 1.4
Manipulator weight (kg)	Link 1 and Dynamixel MX106	0.45
	Link 2 and Dynamixel MX64	0.30
	Link 3, Dynamixel MX64, and actuator	0.50
Manipulator length (m)	Link 1	0.25
	Link 2	0.15
	Link 3	0.15
	Link 4	0.05
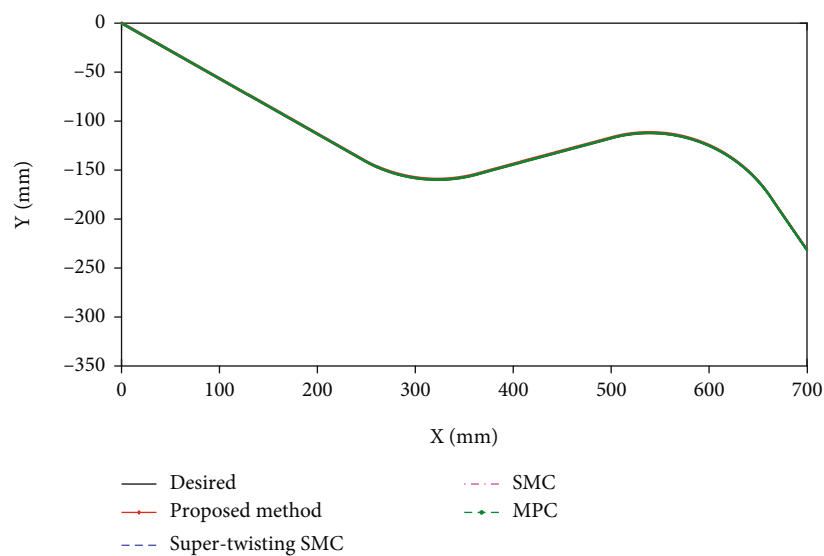
FIGURE 6: Comparison of tracking results in Case 1.
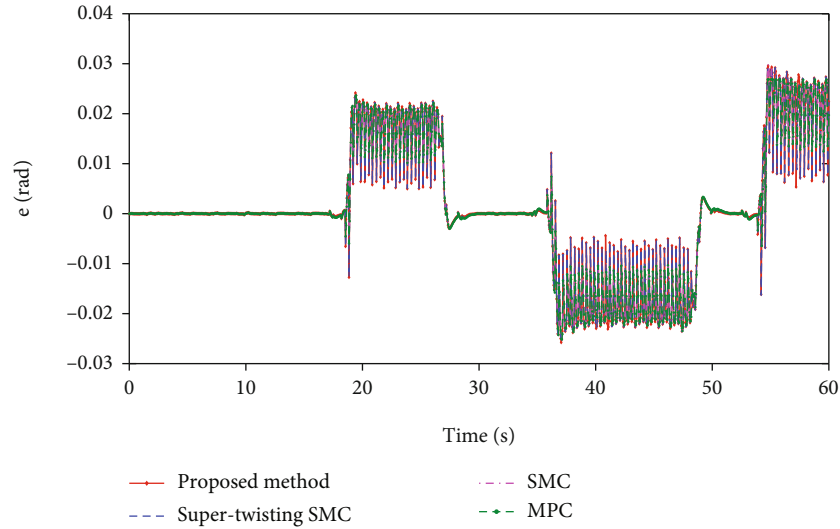
FIGURE 7: Comparison of tracking errors in Case 1.
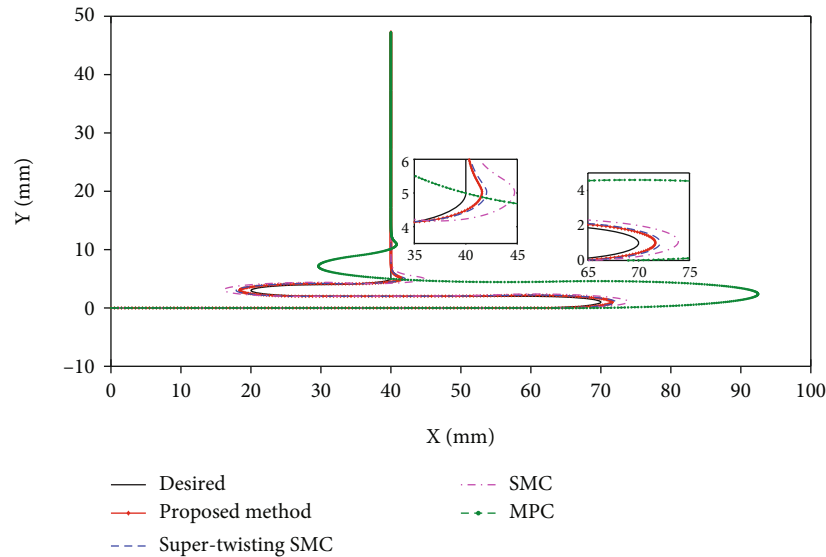
FIGURE 8: Comparison of tracking results in Case 2.

predicted slip vector can be easily calculated by (36). The same method applies to each joint.

4. Simulation Results

In this part, the manipulator model is established by using MATLAB 2021a, and the designed controller is simulated. The simulated manipulator adopts the practical 4-DOF manipulator. In the simulation of this paper, fixed kinematic parameters are assumed, but we introduce serious 15% uncertainty in other parameters, that is, connecting link mass, connecting link inertia, and connecting link centroid position, and periodically introduce 20% torque error to simulate periodic collision caused by human-robot cooperation. The structure diagram of the controller is shown in Figure 5.

In the simulation part, we compare the proposed scheme with sliding mode control [30], supertwisting sliding mode control [31], and model predictive control [33], which are common methods used in manipulator trajectory tracking control. And we design two case examples to illustrate the application effect of the algorithm under different conditions. In the simulation, the controller parameters and manipulator mechanical parameters of the scheme proposed in this paper are shown in Table 3

Case 1. In this case, we designed a large-area cutting scheme of the manipulator. There are no sharp turns in the whole track. In order to respond to the interference of human-robot cooperation, a periodic external force is added at the second joint, and the trajectory tracking results are shown in Figure 6.
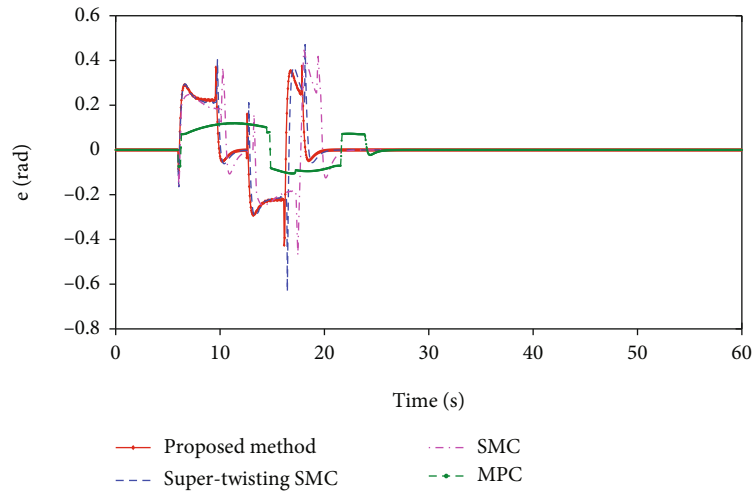
FIGURE 9: Comparison of tracking errors in Case 2.
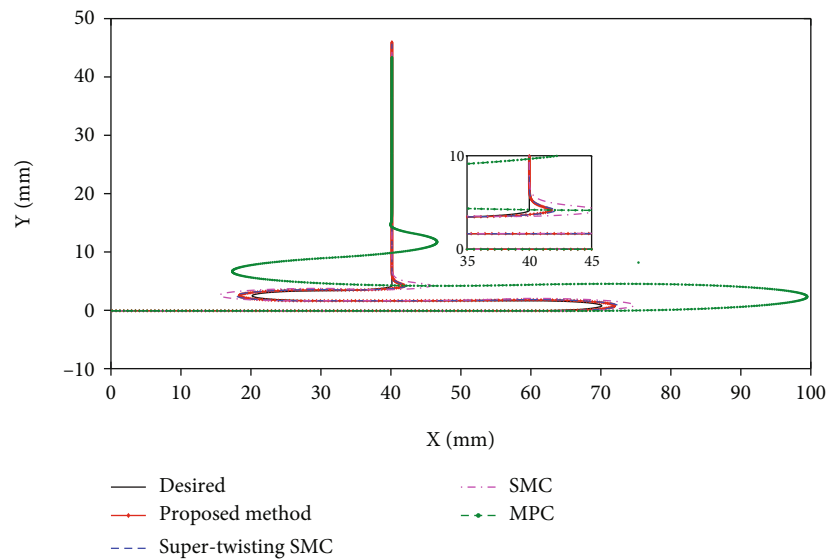
FIGURE 10: Comparison of tracking results in Case 3.

From the track tracking effect shown in Figure 6 and the tracking error shown in Figure 7, it can be seen that the adopted method does not cause obvious error under the influence of periodic external force, which shows that these classical algorithms have strong robustness, and the selected comparison scheme is suitable for manipulator control.

Case 2. In this case, we designed the cutting work of the small area sharp turning process of the manipulator. There are many turning points in the trajectory, but no external force interference is added. The track tracking results are shown in Figure 8.

As can be seen from Figure 8, the method proposed in this paper can track the desired trajectory faster than other methods after the turning of the trajectory. This shows that the proposed method still has performance advantages in trajectory tracking control even without direct external force. As shown in Figure 9, the method proposed in this

paper has an advantage in convergence speed, and the trajectory can track the desired trajectory quickly after errors arise from uncertain disturbances and acute trajectories.

Case 3. In this case, the trajectory of the manipulator is the same as that in Case 2. At the same time, the same periodic external force as in Case 1 is added. The track tracking results are shown in Figure 10.

From the tracking effect of model predictive control in Figure 10, it can be seen that the error with the desired trajectory is larger than that in Figure 8, which shows that the influence of the applied external force on trajectory tracking is obvious. For other sliding mode control methods, due to the model prediction of stiffness matrix, it can actively compensate the influence caused by the external force. Therefore, it can be seen in Figure 11. In terms of error and convergence time, there is little difference between several sliding mode controls and Figure 9, which reflects the
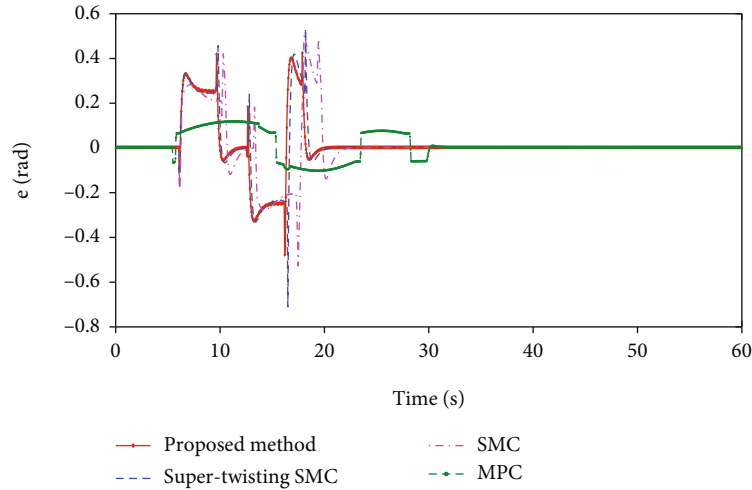
Figure 11: Comparison of tracking results in Case 3.
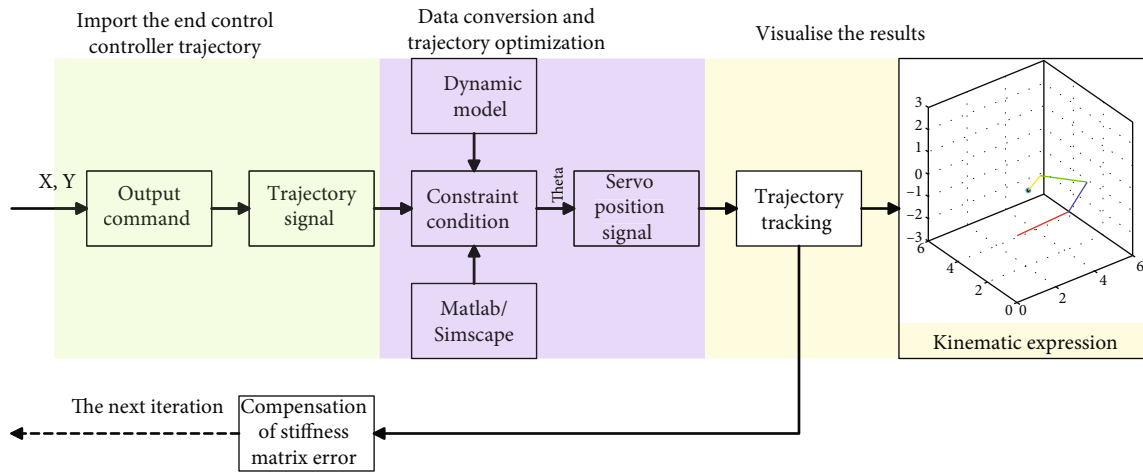
Figure 12: Simulation process of trajectory tracking control in Simscape.

effectiveness of the joint stiffness model prediction proposed in this paper.

It should be noted that since the tracking effect of the model predictive control has a large error, in order to describe the convergence process of the control method simultaneously with other methods, in Figures 9 and 11, the error of the model prediction is based on the deviation between the interpolation projection points of the nearest sampling point and the next sampling point.

In order to illustrate the performance of the scheme proposed in this paper in practical application in manipulator as much as possible, a 4-DOF manipulator model is established by using MATLAB/Simscape, and the external force is added at the joint to verify the performance of the scheme in the human-robot cooperation application of the assembly line. Because Simscape involves many variables and the coupling between parameters is strong, the simulation needs a lot of calculation. With the permission of computer performance and time, we will increase the number of simulation iterations as much as possible. The following figure shows the simulation results of the 10th iteration.

Firstly, the coordinate data of the desired trajectory of the manipulator in the motion space are read, and the constraints of the motion process are fitted according to the physical model established based on the mechanical characteristics of the manipulator. Secondly, the stiffness identification model is used to process the trajectory deviation caused by external force, which is converted into servo motor control signal and provided to the joint controller. Finally, the motion process obtained after Simscape simulation is visualized in MATLAB, and the proposed stiffness matrix compensation method will be used to correct the error before entering the next iteration. The specific process described above is shown in Figure 12.

The experiment of two closed curves is designed according to the process in Figure 12. In the application process of industrial manipulator, the requirements of closed pattern are high. It needs to overlap the starting point and terminal point, and this standard should still be maintained after repeated operations; otherwise, waste workpieces or unqualified products will be produced. The experiments of the two closed curves shown in Figure 13 take the effect after the

(a)

(b)
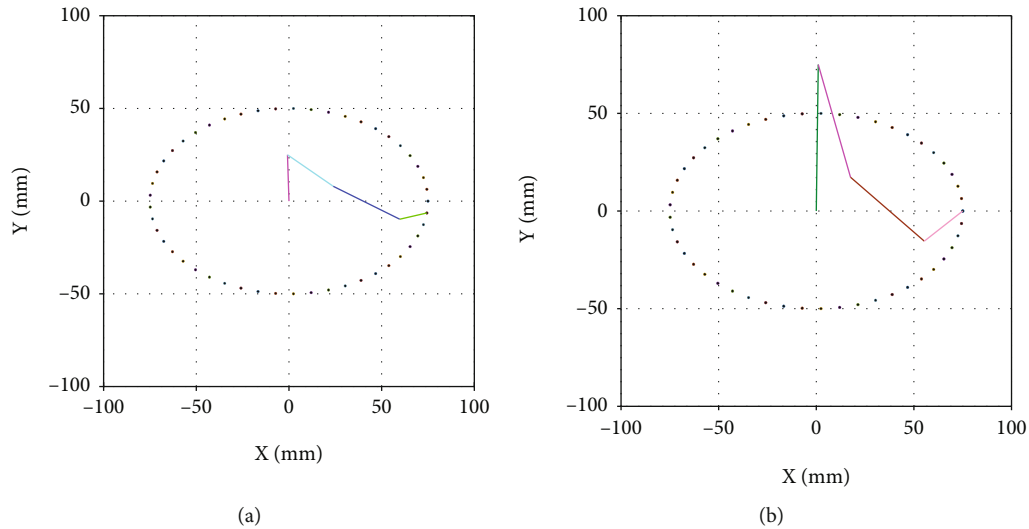
Figure 13: Simulation results in Simscape.

10th iteration, and the applied periodic external force is 20 ± 2 N. The difference between the two groups of experiments is that the length of the connecting rod of the manipulator is different, but as shown in Figures 13(a) and 13(b), the starting point and end point can be kept coincident, which shows that the specification of the manipulator has little impact on the application effect of the scheme proposed in this paper.

5. Conclusion

In this paper, the arctangent terminal sliding mode surface and sliding mode controller are designed for the manipulator with fixed periodic external force on the assembly line applied to human-robot cooperation, so as to track the reference trajectory quickly and accurately. In order to suppress the influence of uncertain disturbance on the controller, the fuzzy logic adaptive parameters are constructed, combined with sliding mode control to enhance the robustness in the control process. Then, the supertwisting and fuzzy adaptive theory are used to reduce the chattering and compensate the concentrated disturbance of the manipulator under the action of external force. Then, because the sensor has some limitations in stiffness measurement, this paper designs a stiffness identification model to estimate the disturbance caused by the applied external force, which saves the hardware cost of the sensor and improves the control effect. The feasibility of the proposed FASTSMC is verified by the simulation of 4-DOF manipulator system.

In the future, we will study the trajectory tracking control of the manipulator under the influence of external forces with more complex characteristics, so as to reduce the torque amplitude and disturbance on the premise of ensuring the convergence speed. In addition, the stiffness identification method proposed in this paper is only limited to the case that the manipulator receives the influence of strong periodic external force, which has certain limitations in practical applications. In the follow-up research, we will study a reasonable data-driven method and combine the end trajectory offset caused by deformation law with stiffness identification to improve the robustness of the controller.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] A. K. Inkulu, M. V. A. R. Bahubalendruni, A. Dara, and K. SankaranarayanaSamy, "Challenges and opportunities in human robot collaboration context of Industry 4.0-a state of the art review," *Industrial Robot: the International Journal of Robotics Research and Application*, vol. 49, 2021.

[2] A. Kolbeinsson, E. Lagerstedt, and J. Lindblom, "Foundation for a classification of collaboration levels for human-robot cooperation in manufacturing," *Production & Manufacturing Research*, vol. 7, no. 1, pp. 448–471, 2019.

[3] S. Kizir and A. Elşavi, "Position-based fractional-order impedance control of a 2 DOF serial manipulator," *Robotica*, vol. 39, no. 9, pp. 1560–1574, 2021.

[4] D. Reyes-Uquillas and T. Hsiao, "Safe and intuitive manual guidance of a robot manipulator using adaptive admittance control towards robot agility," *Robotics and Computer-Integrated Manufacturing*, vol. 70, p. 102127, 2021.

[5] G. Boucher, T. Laliberté, and C. Gosselin, "A parallel low-impedance sensing approach for highly responsive physical human-robot interaction," *International Conference on Robotics and Automation*, 2019, Montreal, QC, Canada, May 2019, 2019.

[6] J. Li, W. Zhang, and Q. Zhu, "Weighted multiple-model neural network adaptive control for robotic manipulators with jumping parameters," *Complexity*, vol. 2020, 12 pages, 2020.

[7] A. R. Reina, K.-D. Nguyen, and H. Dankowicz, "Experimental validation of an adaptive controller for manipulators on a dynamic platform," *Robotica*, vol. 39, no. 4, pp. 582–605, 2021.

[8] Y. She, S. Li, and M. Xin, "Quantum-interference artificial neural network with application to space manipulator control," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 57, no. 4, pp. 2167–2182, 2021.

[9] D. Wei, T. Gao, X. Mo, R. Xi, and C. Zhou, "Flexible biotensegrity manipulator with multi-degree of freedom and variable structure," *Chinese Journal of Mechanical Engineering*, vol. 33, no. 1, pp. 1–11, 2020.

[10] Y. Deng, "Adaptive finite-time fuzzy command filtered controller design for uncertain robotic manipulators," *International Journal of Advanced Robotic Systems*, vol. 16, no. 1, p. 172988141982814, 2019.

[11] C. Figueredo and L. Felipe, "Robust H∞ kinematic control of manipulator robots using dual quaternion algebra," *Automatica*, vol. 132, p. 109817, 2021.

[12] S. Yi and J. Zhai, "Adaptive second-order fast nonsingular terminal sliding mode control for robotic manipulators," *ISA Transactions*, vol. 90, pp. 41–51, 2019.

[13] M. Nasiri, S. Mobayen, and A. Arzani, "PID-type terminal sliding mode control for permanent magnet synchronous generator based enhanced wind energy conversion systems," *CSEE Journal of Power and Energy Systems*, 2021.

[14] X. Zhang and R. Shi, "Research on adaptive non-singular fast terminal sliding mode control based on variable exponential power reaching law in manipulators," *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 236, no. 3, pp. 567–578, 2022.

[15] T. N. Truong, A. T. Vo, and H.-J. Kang, "A backstepping global fast terminal sliding mode control for trajectory tracking control of industrial robotic manipulators," *IEEE Access*, vol. 9, pp. 31921–31931, 2021.

[16] T. Rojsiraphisal, S. Mobayen, J. H. Asad, A. Chang, and J. Puangmala, "Fast terminal sliding control of underactuated robotic systems based on disturbance observer with experimental validation," *Mathematics*, vol. 9, 2021.

[17] J. Zhai and X. Gui, "A novel non-singular terminal sliding mode trajectory tracking control for robotic manipulators," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 68, no. 1, pp. 391–395, 2021.

[18] D.-T. Tran, H.-V.-A. Truong, and K. K. Ahn, "Adaptive nonsingular fast terminal sliding mode control of robotic manipulator based neural network approach," *International Journal of Precision Engineering and Manufacturing*, vol. 22, no. 3, pp. 417–429, 2021.

[19] M. Mazare, M. Taghizadeh, and P. Ghaf-Ghanbari, "Fault-tolerant control based on adaptive super-twisting nonsingular integral-type terminal sliding mode for a delta parallel robot," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 42, no. 8, pp. 1–15, 2020.

[20] S. Mobayen, A. Fekih, S. Vaidyanathan, and A. Sambas, "Chameleon chaotic systems with quadratic nonlinearities: an adaptive finite-time sliding mode control approach and circuit simulation," *IEEE Access*, vol. 9, pp. 64558–64573, 2021.

[21] K. A. Alattas, S. Mobayen, W. Assawinchaichote et al., "A Lyapunov-based optimal integral finite-time tracking control approach for asymmetric nonholonomic robotic systems," *Symmetry*, vol. 13, no. 12, p. 2367, 2021.

[22] M. Boukattaya, N. Mezghani, and T. Damak, "Adaptive nonsingular fast terminal sliding-mode control for the tracking problem of uncertain dynamical systems," *ISA Transactions*, vol. 77, pp. 1–19, 2018.

[23] S.-J. Huang, Y.-C. Liu, and S.-H. Hsiang, "Robotic end-effector impedance control without expensive torque/force sensor," *International Journal of Mechanical and Mechatronics Engineering*, vol. 7, no. 7, pp. 1446–1453, 2013.

[24] A. Mitra and L. Behera, "Development of a fuzzy sliding mode controller with adaptive tuning technique for a mri guided robot in the human vasculature," in *IEEE 13th International Conference on Industrial Informatics*, Cambridge, UK, July 2015.

[25] L. Han, G. Tang, Z. Zhou, H. Huang, and D. Xie, "Adaptive wave neural network nonsingular terminal sliding mode control for an underwater manipulator with force estimation," *Transactions of the Canadian Society for Mechanical Engineering*, vol. 45, no. 2, pp. 183–198, 2021.

[26] Z. Ma and G. Dong, "Variable stiffness and damping impedance control with disturbance observer," in *7th International Conference on Control, Mechatronics and Automation*, Delft, Netherlands, November 2019.

[27] A. Fayazi, N. Pariz, A. Karimpour, and S. H. Hosseinnia, "Robust position-based impedance control of lightweight single-link flexible robots interacting with the unknown environment via a fractional-order sliding mode controller," *Robotica*, vol. 36, no. 12, pp. 1920–1942, 2018.

[28] C. Chen, C. Zhang, T. Hu, H. Ni, and Q. Chen, "Finite-time tracking control for uncertain robotic manipulators using backstepping method and novel extended state observer," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, article 1729881419844655, 2019.

[29] Y. J. Ma, H. Zhao, and T. Li, "Robust adaptive dual layer sliding mode controller: methodology and application of uncertain robot manipulator," *Transactions of the Institute of Measurement and Control*, vol. 44, 2022.

[30] W. Ge, D. Ye, W. Jiang, and X. Sun, "Sliding mode control for trajectory tracking on mobile manipulators.," in *APCCAS 2008-2008 IEEE Asia Pacific Conference on Circuits and Systems*, Macao, December 2008.

[31] C.-S. Jeong, J.-S. Kim, and S.-I. Han, "Tracking error constrained super-twisting sliding mode control for robotic systems," *International Journal of Control, Automation and Systems*, vol. 16, no. 2, pp. 804–814, 2018.

[32] S. Fan and S. Fan, "Approximate stiffness modelling and stiffness defect identification for a heavy-load parallel manipulator," *Robotica*, vol. 37, no. 6, pp. 1120–1142, 2019.

[33] K. Belda and M. Tchaikovsky, "Predictive and anisotropic control design for robot motion under stochastic disturbances," in *18th European control conference (ECC).*, Naples, Italy, June 2019.

Hindawi

Research Article

Sensor Selection Scheme considering Uncertainty Disturbance

Shi Wentao [iD],[1] **Chen Dong,**[1] **Zhou Lin** [iD],[2] **Bai Ke,**[2] **and Jin Yong** [iD][2]

[1]*School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an 710072, China*
[2]*School of Artificial Intelligence, Henan University, Zhengzhou 450046, China*

Correspondence should be addressed to Zhou Lin; zhoulin@henu.edu.cn

In multisensor cooperative detection network, some random disturbances, energy carried by sensor, distance between target and sensor node, and so on all affect the sensor selection scheme. To effectively select some sensors for detecting the target, a novel sensor selection method considering uncertainty disturbance is proposed under constraints of estimation accuracy and energy consumption. Firstly, the sensor selection problem is modeled as a binary form optimization problem with a penalty term to minimize the number of sensors. Secondly, some factors (precision, energy, and distance, etc.) affecting the sensor selection scheme are analyzed and quantified, and energy consumption matrix and estimation precision threshold are given by matrix tra'nsformation. Finally, the problem of minimizing sensor number after relaxation is solved by convex optimization method, obtaining sensor selection scheme by discretization and legitimization of the suboptimal solution after convex relaxation. Simulation results show that the proposed algorithm can ensure the minimum number of sensors, improving accuracy of state estimation and saving network energy.

1. Introduction

Wireless Sensor Network (WSN) is a self-organization wireless network composed of large numbers of microsensor nodes. Due to it being easy to deploy and inexpensive, WSN widely is applied in areas such as the military, aviation, and agriculture [1]. At the same time, all sensor nodes in WSN can be used to obtain the most comprehensive information related to target; however, it leads to the energy consumption and processing burden of the network. Thus, according to measurement accuracy requirements, selecting part sensor nodes can save network energy, reduce response delay, and prolong the lifetime of the network [2].

1.1. Related Works. Some researches of sensors selection have been applied in target tracking system, cooperative detection system, and so on [3, 4]. Jamali et al. embedded a penalty item into objective function of sensors selection to enhance sparsity of solver, and it was only used in low dimension scene [5]. A greedy algorithm called FrameSense was proposed by Ranieri et al., and it iteratively removed the

sensor corresponding to the maximum cost function [6]. FrameSense's algorithm could obtain optimal sensor selection scheme, but heavy compute burden and multiple constraint conditions were needed. Minimum eigenspace method was proposed by Jiang et al., which could quickly converge to obtain sensor selection scheme; however, lower estimation accuracy was its disadvantage [7], so an improved minimum eigenspace method with group greedy scheme was proposed by Jiang et al. to improve about 63% estimation accuracy expect heavy compute burden [8].

In spite of sensor selection can be modeled as integer programming, it is NP-hard problem. Branch and bound (BBD) was an effective global optimal method corresponding to integer programming; however, computation complexity was exponentially growing with increase of dimension corresponding to optimized variable [9]. Based on convex method, Joshi and Boyd proposed a heuristic method to approximately obtain sensor selection scheme [10]. Since the constraint of the problem was relaxed, Joshi and Boyd's method could reduce the accuracy of measurement. So, some local optimum techniques were proposed by Joshi to enhance accuracy corresponding to sensor

selection problem, while it had good detection performance with a small number of sensors.

Besides, communication performance and remaining energy of sensors are two key factors in multisensor cooperative detection network, and they should be considered in sensor selection too. Sensor selection technique with time, energy, and communication constraints was discussed by Rusu et al. [11]; they modeled the sensor selection problem as a 0-1 optimization problem, then used a convex relaxation method to solve the objective function. However, the influence of some factors (sensor measurement capability, sensor location, etc.) on sensor scheduling was not involved in Rusu et al.'s paper. In addition, some various factors (uncertainty disturbance, vibration disturbance, etc.) caused by complexity environment may decrease measurement accuracy of selected sensors. So, it is important that selecting suitable sensors under some constraints including measurement accuracy, energy consumption and communication capability.

1.2. Contributions. Under some conditions including bounded disturbance of sensor measurement and sensor location, a new sensor selection algorithm based on the constraints of energy and estimation accuracy is proposed. By using convex relaxation technology, the minimum number of sensor selection set is obtained, and it balances network energy and measurement accuracy to ensure the performance of the multisensor cooperative detection system. The main contributions are as follows:

(1) A binary form optimization model with penalty term is proposed to obtain the sensor selection scheme, which is used to minimize number of sensors. Comparing with tradition algorithms, our energy penalty term is used to balance energy consumption

(2) The distance weight between sensor and target is adaptively given. It directly reflects that the closer the sensor is to target, the greater probability of being selected, and the better detection accuracy of sensor. So, compared with the traditional method, the number of selected sensors can be effectively reduced

(3) The energy consumption matrix and state estimation precision are regarded as constraints of the sensor selection problem; it more reflects various factors of the complex environment

(4) We introduce techniques including the convex relaxation, discretization, and legitimization of bounded interval [0,1], which are used to handle Np-hard problem of sensor scheduling and receive suboptimal solution. The proposed method has lower computation complexity than traditional methods

1.3. Paper Organization and Notation. The rest of this paper is organized as follows. Section 2 is problem statement for knowledge research background. Section 3 proposes the improved method for the sensor selection scheme under various constraints including precision, energy, and distance. In this section, sensor selection with penalty term, sensor selection considering uncertainty disturbance, optimization of sensor selection, and description of the proposed algorithm are given separately. Section 4 shows simulation results and analysis. Some conclusions about this paper are given in Section 5.

The symbol denotement of this paper is described as shown in Table 1.

2. Problem Statement

Consider a target $\mathbf{x}$ with N-dimensional state is cooperatively detected by sensor set Ω which composed of M sensors. Measurement matrix Φ is related to set Ω, which can be defined as $\Phi = [\varphi_1, \varphi_2, \cdots, \varphi_M]^T \in R^{M \times N}, M > N$, and $\varphi_i \in R^{N \times 1}$ is the measurement vector corresponding to sensor i. At a sampling time t, k_t sensors are selected to form measurement sensor set Ω_{kt} ($\Omega_{kt} \subseteq \Omega$); at the same time, measurement of target state $\mathbf{x}_t$ is described as follows:

$$\mathbf{y}_t = \Phi_t \mathbf{x}_t + v_t, \tag{1}$$

where $\Phi_t \in R^{k_t \times N}, k_t > N$ is the measurement matrix and $\mathbf{y}_t \in \mathbf{R}^{k_t}$ is the measurement of sensors.

Therefore, the minimum unbiased estimate of target state $\mathbf{x}_t$ is $\hat{\mathbf{x}}_t = \Phi_t^\dagger \mathbf{y}_t = (\Phi_t^T \Phi_t)^{-1} \Phi_t^T \mathbf{y}_t$, and its mean square error (MSE) can be expressed as [12]

$$\text{MSE}(\hat{\mathbf{x}}_t) = \text{tr}\left(\left(\Phi_t^T \Phi_t\right)^{-1}\right) = \sum_{i=1}^{n} \frac{1}{\lambda_i\left(\Phi_t^T \Phi_t\right)}. \tag{2}$$

3. The Proposed Algorithm for Sensor Selection

To solve the problem of sensor selection under some constraints, the optimization problem of sensor selection is given under some constraints including environment, energy, and distance between node and target. After being quantized, constraints can be described with the energy consumption matrix and state estimation precision to be regarded as constraints of the optimization problem. Besides, the distance weight between the sensor and target is adaptively given to reflect that the closer the sensor is to target, the greater the probability of being selected. Convex relaxation methods are considered to handle optimization problem of sensor selection, and discretization and legitimization of suboptimal solution after convex relaxation are used to obtain the suboptimal solution of the sensor selection scheme.

3.1. Problem of Sensor Selection. Obviously, state estimation accuracy of target $\mathbf{x}$ is the highest $\gamma_0 = \text{tr}((\Phi^T \Phi)^{-1})$ when sensor set $\Omega_{kt} = \Omega$, i.e., all sensors of sensor cooperative detection system are used to detect target. However, it consumes a lot of energy; moreover, there are some uncertain environment factors such as temperature, humidity, and visibility, which reduce the estimation accuracy of the target state. Therefore, considering environment factors and estimation performance, the problem of sensor selection can

TABLE 1: Symbol denotement.

Symbol	Denotement
Φ (or ν)	Measurement matrix (vector)
$(\cdot)_i$	i-th element of vector
$\mathbf{1}$	Vector with proper dimension whose entry is 1
diag $(\cdot)$	Diagonal of matrix
card $(\cdot)$	Cardinal of given set
$\lambda_i(\cdot)$	i-th eigenvalue of matrix
$(\cdot)^T$	Transposition of matrix
$(\cdot)^{-1}$	Inverse of matrix
tr$(\cdot)$	Trace of matrix
$\odot$	Hadamard product
$\|\cdot\|_2$	2-Norm
$\|\cdot\|_\mathbf{F}$	$\mathbf{F}$-Norm
$\|\cdot\|_\infty$	∞-Norm
t	Sampling time
$\mathbf{x}_t$	State of target $\mathbf{x}$ sampling time t
k_t	Number of sensors
Φ_t	Composed of k_t rows in matrix Φ
ν_t	Zero mean Gaussian noise with $\sigma^2\mathbf{I}$ covariance
$\mathbf{y}_t$	Measurement of sensor
$\mathbf{b}$	Coordinate of target $\mathbf{x}$

be expressed as the minimum cardinal of sensor measurement set Ω_{kt} as follows:

$$
\begin{aligned}
\underset{\Omega_{kt}}{\text{minimize}} \quad & \sum_{t=1}^{T} \text{card}(\Omega_{kt}), \\
\text{s.t.} \quad & r\left(\left(\Phi_t^T \Phi_t\right)^{-1}\right) \le \tau\gamma_0 - C, \\
& C - \gamma_0(\tau - 1) \le 0,
\end{aligned}
\tag{3}
$$

where card $(\cdot)$ is the function of minimum cardinal. T is the total sampling time. $\tau > 1$ is the accuracy level which increases with the decrease of accuracy requirement. $C = V/(pht_0)$ (V, p, h, t_0 express visibility, environmental factor parameter, humidity, and temperature, respectively) is the environment factor. When τ increases, it means that fewer sensors can be used to meet estimated accuracy requirement; besides, inequality $C - \gamma_0(\tau - 1) \le 0$ is used to restrict the estimation accuracy which does not exceed the optimal estimation accuracy.

In short, equation (3) indicates that we want to choose a sensor set with minimum cardinal satisfying estimation accuracy constraint conditions. To describe the problem of sensor selection, M-dimensional vector $\omega_t \in \{0, 1\}^M$ is introduced to express the selected state of sensors in sensor set Ω at t sampling time, and $(\omega_t)_i$ encodes whether the i-th sensor is selected. The selected state of sensors at all sampling time T can be described by matrix $\Theta = [\omega_1, \omega_2, \cdots, \omega_T]$. Thus, the optimization problem of sensor selection described in equation (3) can be transformed into the following equation:

$$
\begin{aligned}
\underset{\Theta \in \{0,1\}^{M \times T}}{\text{minimize}} \quad & \sum_{t=1}^{T} \alpha_t^T \omega_t, \\
\text{s.t.} \quad & \text{tr}\left(\left(\Phi^T \text{diag}(\omega_t)\Phi\right)^{-1}\right) \le \tau\gamma_0 - C, \\
& C - \gamma_0(\tau - 1) \le 0,
\end{aligned}
\tag{4}
$$

where weight vector $\alpha_t \in R^M$ and $(\alpha_t)_i = ((\omega_t)_i + \epsilon)^{-1}$ ($i = 1, \cdots, M$); ϵ is a small number to avoid denominator 0 during iterative processing.

3.2. Optimization Model with Penalty Term. As it is known, energy consumption of sensor is positively related to distance between sensor and target, at the same time, some sensors die after repeated use, shortening network lifetime. So, it is important that considering the balance of energy consumption to avoid some selected sensors from dying too quickly. Here, we introduce sensor usage factor into problem of sensor selection to constrain energy consumption of selected sensors.

First of all, considering each sensor to be selected at least once for detecting target, we add equation (5) into equation (4) and regard it as the constraint of optimization problem of sensor selection

$$
\sum_{t=1}^{T} \omega_t \ge \mathbf{1}.
\tag{5}
$$

Then, to increase the measurement efficiency, some sensors near the target should be selected with high probability. Therefore, the constraint condition (4) in equation (4) can be modified as follows:

$$
\text{tr}\left(\left(\Phi^T \text{diag}(\mathbf{H})\Phi\right)^{-1}\right) \le \tau\gamma_0 - C,
\tag{6}
$$

where the weight vector of distance is $\mathbf{H} = (S_M \omega_t) \odot \mathbf{d}_M$, and $(\mathbf{d}_M)_i = 1/\|\mathbf{m}_i - \mathbf{b}\|_2$, $S_M = SC/M$. S_M is a relative area by monitoring area S, environmental factor C, and number of sensor nodes M.

Next, considering the measurement and processing cost of sensors, we add equation (7) into the constraint condition of equation (4).

$$
(\text{diag}(\mathbf{s})\Theta\mathbf{1}) \odot \mathbf{P}_L \le \mathbf{e_0} + \mathbf{E},
\tag{7}
$$

where $\mathbf{s} = [\|\varphi_1\|_2^2, \|\varphi_2\|_2^2, \cdots, \|\varphi_M\|_2^2]^T \in R_+^M$ is the cost of measurement and data processing about all selected sensors, and $(\mathbf{s})_i$ is the cost about the i-th sensor. $(\mathbf{P}_L)_i = (\mathbf{d}_M)_i^\eta$ represents the distance-dependent Rayleigh channel attenuation of the i-th sensor, and η is the channel attenuation factor. $\mathbf{e_0} \in R_+^M$ is the available reference energy of the sensor, and $(\mathbf{e_0})_i$ is the energy about the i-th sensor; besides, the extra energy beyond the reference energy is expressed by $\mathbf{E} \in R^M$.

Obviously, to avoid excessive energy consumption of sensors, it is necessary to add a penalty term $g(\mathbf{E})$ relating to the extra energy in equation (4); therefore, it is revised as follows:

Input: initialization parameters.
Output: sensor scheduling matrix $\Theta \in \{0, 1\}^{M \times T}$.
1: Setting initial weight $\alpha_t = \mathbf{1}$, sensor scheduling matrix $\Theta_0 = \mathbf{0}_{M \times T}$.
2: Initializing unselected sensor node set $N = \varnothing$, and selected node set $K = \varnothing$.
3: Calculating minimum MSE γ_0.
4: FOR st: =1 to Maxst DO
5:    Obtaining current Θ by solving optimization problem (14) based on Θ_0Maxst.
6:    Updating set $N = \{n \mid \Theta(n) \leq \epsilon\}$ and $K = \{k \mid \Theta(k) \geq 1 - \epsilon\}$.
7:    IF $\|\Theta - \Theta_0\|_F^2 \leq \epsilon$ THEN
8:      $K \longleftarrow K \cup \{\arg\max_k \Theta(k), k \notin K\}$.
9:    ELSE IF $|\mathbf{N}| + |\mathbf{K}| = M \times T$ THEN
10:     Going to step 15.
11:   ELSE
12:     Setting sensor scheduling matrix $\Theta_0 \longleftarrow \Theta$, and updating weights based on $(\alpha_t)_i \longleftarrow ((\omega_t)_i + \epsilon)^{-1}$, then going to step 5.
13:   END IF
14: END FOR
15: Getting sensor scheduling matrix Θ.

ALGORITHM 1: The pseudocode of proposed algorithm.

$$\underset{\mathbf{E} > \mathbf{0}, \Theta \in \{0,1\}^{M \times T}}{\text{minimize}} \quad \sum_{t=1}^{T} \alpha_t^T \omega_t + \lambda g(\mathbf{E}),$$

$$\text{s.t.} \quad \text{tr}\left(\left(\Phi^T \text{diag}(\mathbf{H}) \Phi \right)^{-1} \right) \leq \tau \gamma_0 - C,$$

$$C - \gamma_0(\tau - 1) \leq 0, \tag{8}$$

$$(\text{diag}(\mathbf{s}) \Theta \mathbf{1}) \odot \mathbf{P}_L \leq \mathbf{e_0} + \mathbf{E},$$

$$\sum_{t=1}^{T} \omega_t \geq \mathbf{1},$$

where penalty term $g(\mathbf{E}) = \|\mathbf{E}\|_2^2$, λ is a regularization coefficient.

3.3. Sensor Selection considering Uncertainty Disturbance. Various uncertainty disturbances may refluent the measurement performance and position of the sensor; thus, we assume that the upper bounds of measurement disturbance and the channel attenuation disturbance are δ_Φ and δ_{PL}, respectively. Based on the above upper bounds, equation (8) can be transformed as follows:

$$\underset{\substack{\mathbf{E} > \mathbf{0}, \Theta \in \{0,1\}^{M \times T} \\ \|\Delta_\Phi\|_2 < \delta_\Phi, \|\Delta_{PL}\|_\infty < \delta_{pL}}}{\text{minimize}} \quad \sum_{t=1}^{T} \alpha_t^T \omega_t + \lambda g(\mathbf{E}),$$

$$\text{s.t.} \quad \text{tr}\left(\left((\Phi + \Delta_\Phi)^T \text{diag}(\mathbf{H})(\Phi + \Delta_\Phi) \right)^{-1} \right) \leq \tau \gamma_0 - C,$$

$$C - \gamma_0(\tau - 1) \leq 0,$$

$$(\text{diag}(\mathbf{s}) \Theta \mathbf{1}) \odot (\mathbf{P}_L + \Delta_{PL}) \leq \mathbf{e_0} + \mathbf{E},$$

$$\sum_{t=1}^{T} \omega_t \geq \mathbf{1},$$

$$\tag{9}$$

where $\Delta_\Phi \in R^{M \times N}$ and $\Delta_{PL} \in R^{M \times 1}$ are the change of measurement disturbance and channel attenuation disturbance, respectively.

Obviously, the 0-1 constraint $\Theta \in \{0, 1\}^{M \times T}$ is nonconvex; thus, it is necessary to relax the 0-1 constraint to a polyhedral constraint $\Theta \in [0, 1]^{M \times T}$, and equation (9) is rewritten as follows:

$$\underset{\substack{\mathbf{E} > \mathbf{0}, \Theta \in [0,1]^{M \times T} \\ \|\Delta_\Phi\|_2 < \delta_\Phi, \|\Delta_{PL}\|_\infty < \delta_{pL}}}{\text{minimize}} \quad \sum_{t=1}^{T} \alpha_t^T \omega_t + \lambda g(\mathbf{E}),$$

$$\text{s.t.} \quad \text{tr}\left(\left((\Phi + \Delta_\Phi)^T \text{diag}(\mathbf{H})(\Phi + \Delta_\Phi) \right)^{-1} \right) \leq \tau \gamma_0 - C,$$

$$C - \gamma_0(\tau - 1) \leq 0,$$

$$(\text{diag}(\mathbf{s}) \Theta \mathbf{1}) \odot (\mathbf{P}_L + \Delta_{PL}) \leq \mathbf{e_0} + \mathbf{E},$$

$$\sum_{t=1}^{T} \omega_t \geq \mathbf{1},$$

$$\tag{10}$$

where constraint conditions $\|\Delta_\Phi\|_\infty < \delta_\Phi$ and (10) can be transformed into linear matrix inequality form through matrix transformation as follows (see Appendix A):

$$\text{tr}(\mathbf{U}^{-1}) \leq \tau \gamma_0 - C,$$

$$\begin{bmatrix} \Phi^T \text{diag}(\mathbf{H})\Phi - \mathbf{U} - \beta I & \Phi^T \text{diag}(\mathbf{H}) \\ \text{diag}(\mathbf{H})\Phi & \text{diag}(\mathbf{H}) + \left(\dfrac{\beta}{\delta_\Phi^2}\right) \mathbf{I} \end{bmatrix} \succeq 0, \quad \beta \geq 0.$$

$$\tag{11}$$

Besides, constraints $\|\Delta_{PL}\|_\infty < \delta_{PL}$ and (10) can also be

TABLE 2: Parameters of simulation.

Name	Symbol	Value		
		Scene 1	Scene 2	Scene 3
Area	S	$40\,\mathrm{m} \times 40\,\mathrm{m}$	$70\,\mathrm{m} \times 70\,\mathrm{m}$	$100\,\mathrm{m} \times 100\,\mathrm{m}$
Number of sensors	M	100	150	200
Dimensional of target state	N	20	30	40
Position of target	$[b_x, b_y]$	(20, 20)	(35, 35)	(50, 50)
Uncertainty bound of sensor position	δ_{PL}	0.8	1.6	2.4
Level of accuracy of estimation	τ		4	
Attenuation factor of channel	η		-0.5	
Regularization parameter	λ		1000	
Environmental factor	μ		100	
Relative humidity	h		50%	
Temperature	t_0		25°C	
Visibility	V		500 m	
Measure matrix	α		100	
Energy of sensors	e_0		0 J	
Environmental regulation parameter	p		100 m/°C	

transformed into vector form as follows (see Appendix B):

$$(\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot (\mathbf{P}_L + \Pi) \leq \mathbf{e_0} + \mathbf{E}, \tag{12}$$

where

$$\Pi = \left[\underbrace{\delta_{PL}, \delta_{PL}, \cdots, \delta_{PL}}_{M}\right]^T. \tag{13}$$

3.4. Optimization of Sensor Selection. Based on the equations (11) and (12), equation (10) can be rewritten as follows:

$$
\begin{aligned}
\underset{\mathbf{E}>0,\Theta\in[0,1]^{M\times T}}{\mathrm{minimize}} \quad & \sum_{t=1}^{T} \alpha_t^T \omega_t + \lambda g(\mathbf{E}), \\
\text{s.t.} \quad & \begin{bmatrix} \Phi^T \,\mathrm{diag}\,(\mathbf{H})\Phi - \mathbf{U} - \beta I & \Phi^T \,\mathrm{diag}\,(\mathbf{H}) \\ \mathrm{diag}\,(\mathbf{H})\Phi & \mathrm{diag}\,(\mathbf{H}) + \left(\dfrac{\beta}{\delta_\Phi^2}\right)\mathbf{I} \end{bmatrix} \succeq 0, \\
& (\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot (\mathbf{P}_L + \Pi) \leq \mathbf{e_0} + \mathbf{E}, \\
& \mathrm{tr}\left(\mathbf{U}^{-1}\right) \leq \tau\gamma_0 - C, \\
& \beta \geq 0, \\
& C - \gamma_0(\tau - 1) \leq 0, \\
& \sum_{t=1}^{T} \omega_t \geq \mathbf{1}.
\end{aligned}
\tag{14}
$$

Obviously, equation (14) is a standard convex optimization problem of sensor selection, and the relaxed sensor scheduling matrix Θ can be obtained by solving this optimization problem.

Of course, the solution of the relaxed equation (14) cannot directly extend to the original equation (9); thus, we handle this problem by the discretization and legalization method [13].

3.5. Description of the Proposed Algorithm. Algorithm 1 shows the proposed algorithm of sensor selection.

4. Simulation Results and Analysis

To test the performance of the proposed algorithm SS-CEED, we compare it with the existing algorithms Spare-Sense [5], FrameSense [6], CON-REL [10], and SS-TECC [11] in a variety of settings as detailed next. We implement all algorithms in MATLAB and the SDP relaxation scheme via CVX [14]. Besides, all simulations are run on a computer with 1.8 GHz Intel Core i5-8250U CPU and 8.00 GB RAM. We give simulation results and analysis on the distribution and number of sensor node selection, besides the estimation accuracy of algorithms with or without disturbance, which are used to describe the performance of the proposed algorithm.

We assume that M stationary sensor nodes carrying the same energy and one target are randomly deployed in simulation scenes 1, 2, and 3. The parameters with respect to simulations are shown in Table 2; at the same time, for the measurement matrix Φ with constraint $\Phi^T\Phi = \alpha\mathbf{I}$, the selection threshold corresponding to the convex relaxation is set to 0.35 [11].

Figure 1 shows five distributions of sensors and target for the four algorithms in scene 1.

(a) CON-REL

(b) FrameSense

(c) SpareSense

(d) SS-TECC
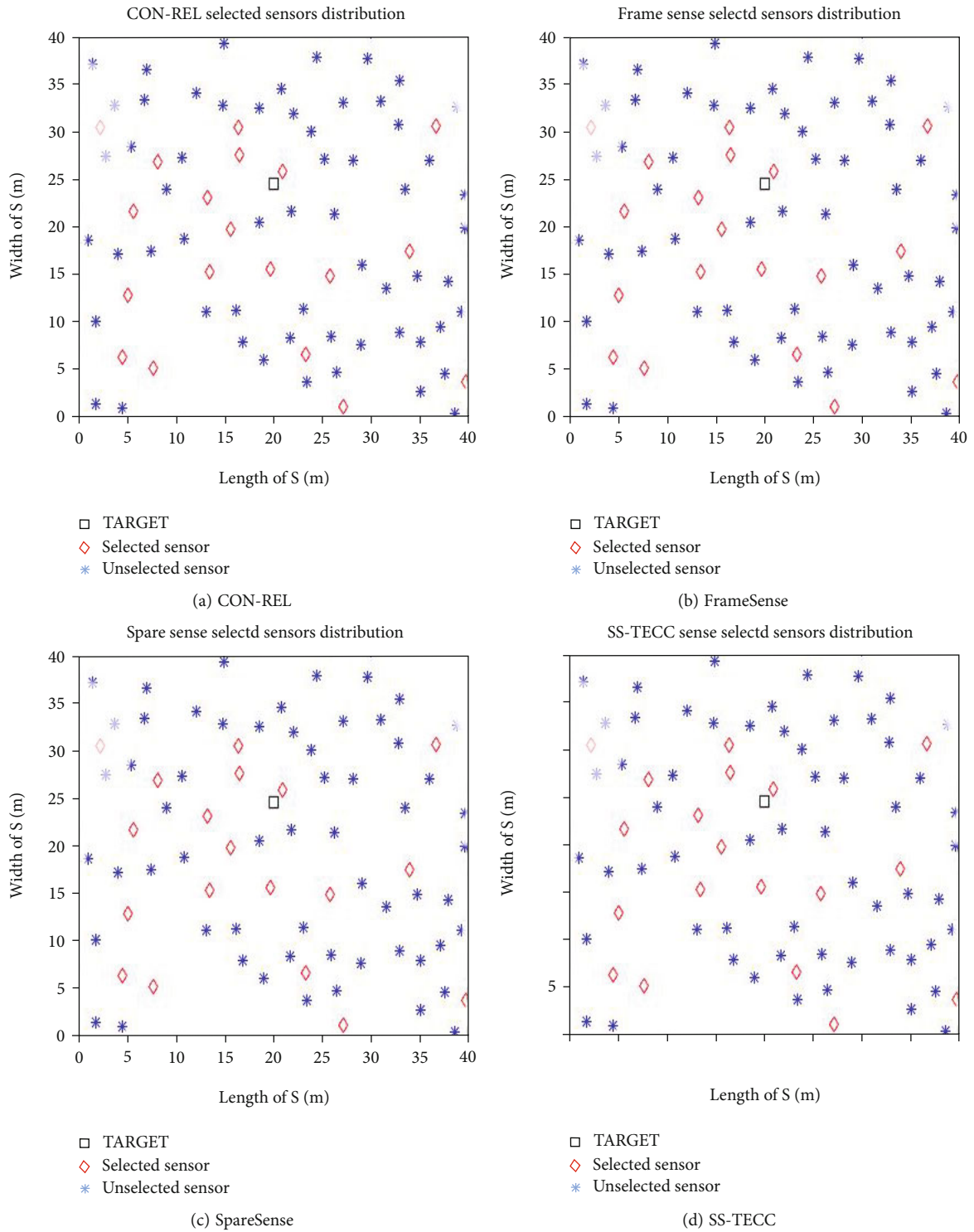
Figure 1: Continued.

(e) SS-CEED
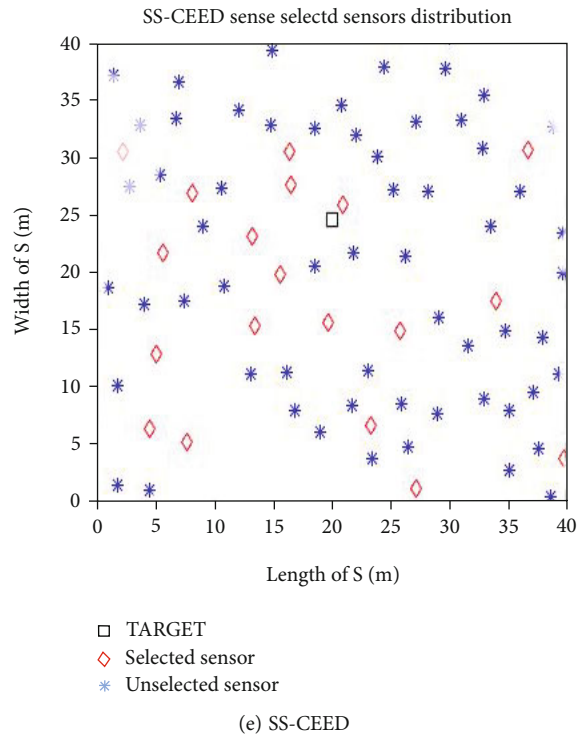
FIGURE 1: Sensor distribution of five algorithms.
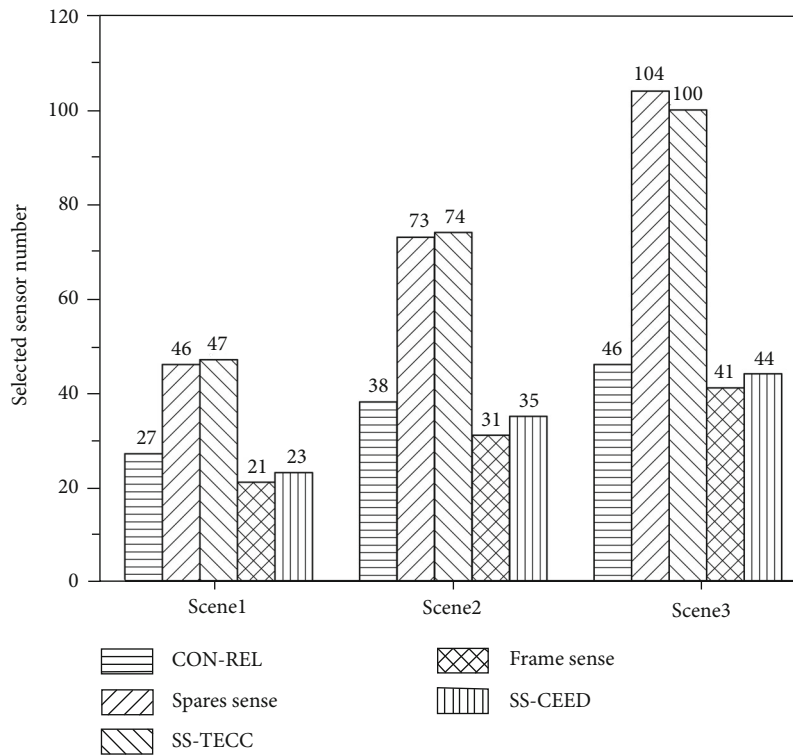
FIGURE 2: Node selection of algorithms under different scenes.

In Figure 1, symbols "□," "◇," and "∗" represent the target, working sensor nodes, and nonworking sensor nodes, respectively. The number of nodes selected by CON-REL, FrameSense, SpareSense, SS-TECC, and SS-CEED is 24, 21, 39, 48, and 21, respectively. It can be found that the least sensor nodes are selected by the proposed SS-CEED algorithm and the FrameSense algorithm, and the number of selected nodes is greater than the dimension of target N
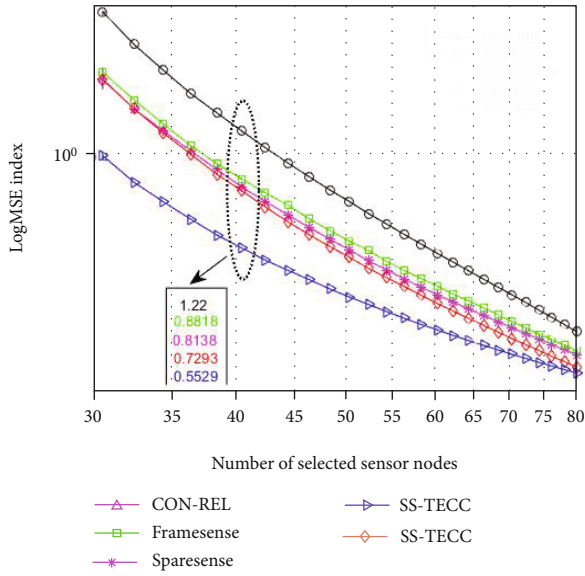
FIGURE 3: MSE comparison of state estimation.

FIGURE 4: WCE comparison of algorithms.

FIGURE 5: The logMSE comparison under different measurement disturbances.
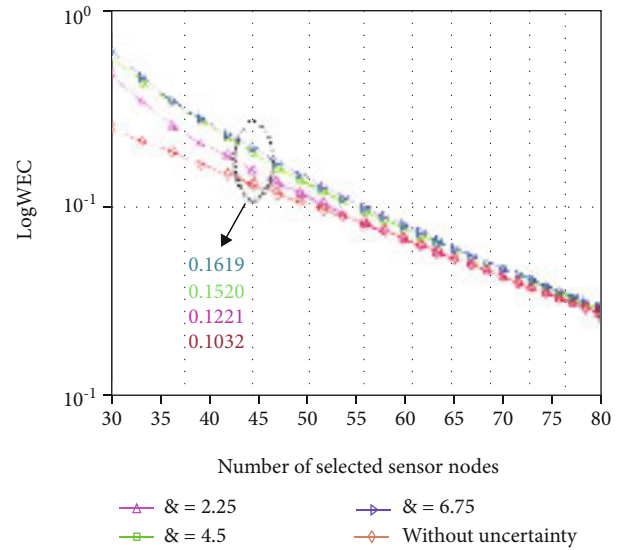
FIGURE 6: The logWCE comparison under different measurement disturbances.

($N = 20$). The reason is that the FrameSense algorithm excludes one node at every time of selecting the senor until the remaining number is $N + 1$, i.e., there have 21 sensors at this time. Other algorithms are similar in selecting sensor nodes. Besides, compared with other algorithms, distance weight is considered in the SS-CEED algorithm, leading to the sensor nodes nearby the target being selected to work; it can effectively decrease energy consumption of communication.

4.1. Analysis of Node Selection. To clearly describe the algorithm performance in different scenes, Figure 2 gives the average number of selected sensors by 100 times Monte Carlo simulation.

In Figure 2, the five algorithms' number of selected sensor nodes increases when the number of deployed nodes increases

in different scenes. The proposed SS-CEED algorithm selects the sensor number as 23, 35, and 44 under three scenes. Compared with other algorithms excluding FrameSense, selecting the sensor number of the proposed SS-CEED algorithm is minimum, and this conclusion is consistent with the analysis result in Figure 1. Although the FrameSense algorithm can select the least number of sensors, it requires that the number of selected sensor nodes must be one more than the dimension of sensor measurement; however, the above constraint is not necessary for the SS-CEED algorithm.

4.2. Analysis of Estimation Error. To show selection performance of five algorithms, this paper gives MSE of target state estimation after deploying selected sensors under different
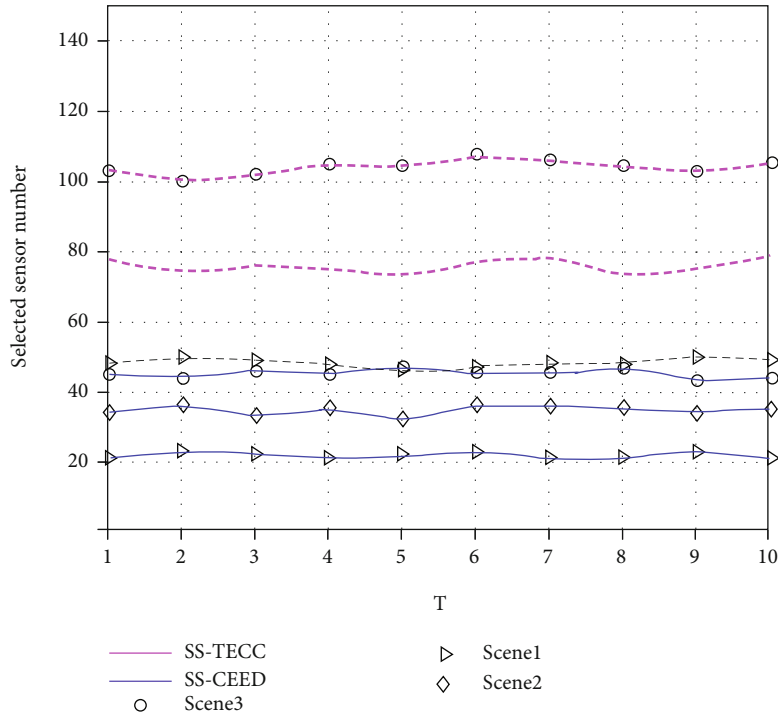
FIGURE 7: Comparison of the number of selected sensors in different scenes.

algorithms. Regarding scene 1 as the simulation background, we compare the MSE and WCE of algorithms under measurement bounded disturbance $\delta_\phi = 1.2$. Figures 3 and 4 give MSE and WCE of algorithms under different selected sensor numbers, respectively.

The logarithm of MSE (logMSE) is displayed to clearly describe the estimation error in Figure 3. It is found that all logMSEs are decreased when the number of selected sensor nodes increase. The reason is that estimation accuracy improves and MSE decreases with the number of sensor nodes used for state estimation increases. In this figure, the logMSE of the proposed SS-CEED algorithm has the smallest value of 0.5529 when the number of selected sensors is 40, and it is 0.1764, 0.2609, 0.3289, and 0.6671 lower than that of SS-TECC, SpareSense, FrameSense, and CON-REL, respectively. The reason is that the SS-CEED algorithm considers measurement and position disturbance of sensors, leading to its estimation performance being the best.

Similar to Figure 3, all WCEs are decreased when the number of selected sensor nodes increases in Figure 4. Compared with SS-TECC, SpareSense, FrameSense, and CON-REL, the proposed SS-CEED algorithm has the smallest WCE, meaning accuracy of state estimation is best.

4.3. Performance Analysis under Different Disturbances. To analyze the influence of different measurement disturbances on the estimation accuracy of the SS-CEED algorithm, this paper gives the result of MSE and WCE after 100 times Monte Carlo simulation under scene 1. Similar to Figure 3, in order to clearly describe the results, the logarithms of MSE and WCE are shown in Figures 5 and 6, named as logMSE and logWCE, respectively.

Assuming the bound measurement disturbances are 0, 2.25, 4.5, and 6.75, Figure 5 shows the logMSE curve of the proposed algorithm. The logMSE of the proposed algorithm is the lowest without measurement disturbance. When there is measurement disturbance, taking 40 sensor nodes as an example, the logMSE with disturbance boundary of 2.25 is 0.0889 smaller than the disturbance boundary of 4.5 and is 0.1157 smaller than the disturbance boundary of 6.75. It is found from different curves of selected sensor nodes. In short, considering measurement disturbance, the estimation accuracy of the proposed algorithm decreases as the measurement disturbance boundary increases.

Similar to logMSE in Figure 5, the logWCE of the proposed algorithm is the lowest without measurement disturbance in Figure 6. And taking 40 sensors as an example, the logWCE with a disturbance boundary of 2.25 is 0.0299 smaller than the disturbance boundary of 4.5 and is 0.0398 smaller than the disturbance boundary of 6.75. Similarly, the estimation accuracy of the proposed algorithm decreases as the measurement disturbance boundary increases under considering measurement disturbance.

4.4. Analysis of Sensor Node Selection in Different Times. Different from other algorithms, the SS-TECC algorithm and the proposed SS-CEED algorithm involve sensor node selection in different working times. In Figure 7, comparison of selection number between the SS-TECC algorithm and the SS-CEED algorithm under all scenes is within $T = 10$ times.

It can be seen from Figure 7, in observation period of the three scenarios, the number of sensors selected by the SS-CEED algorithm and the SS-TECC algorithm changes little; that is, the number of sensor nodes selected by the two

algorithms changes within 5 to save energy. It is obvious that the number of sensors selected by the SS-CEED algorithm is less than that by the SS-TECC algorithm, and the sensor number variation of the SS-CEED algorithm is stationary. In short, compared with the SS-TECC algorithm, the SS-CEED algorithm selects fewer sensors nodes, which can save more network energy.

5. Conclusions

Random disturbances, energy carried by the sensor, distance between the target and the sensor node, and so on affect the performance of sensor selection, so this paper regards the sensor selection problem as a binary optimization problem for sensor scheduling to effectively select sensors to deploy. The optimization problem aimed at the minimum number of selected sensor nodes is constructed, and the constraints such as energy consumption and precision are given. Using matrix transformation and convex relaxation techniques, the optimization problem is solved, and the sensor selection scheme is given. The proposed algorithm not only meets the target state estimation accuracy but also effectively reduces sensor resources and network energy consumption.

Appendix

A. Transformation of Matrix Inequality (11)

Setting $\bar{\Phi} = \Phi + \Delta_\Phi$, $\|\Delta_\Phi\|_2 < \delta_\Phi$, then the optimization constraint (10) can be expressed as

$$
\begin{aligned}
&\mathrm{tr}\left(\mathbf{U}^{-1}\right) \le \tau\gamma_0 - C, \\
&\mathbf{U} \preceq \bar{\Phi}^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\bar{\Phi}, \\
&\|\Delta_\Phi\|_2 < \delta_\Phi,
\end{aligned}
\tag{A.1}
$$

where $\mathbf{U} \in \mathbf{R}^{N \times N}$ is a symmetric positive definite matrix.

Assuming Δ_Φ is bounded, the equation (11) can be transformed into linear matrix inequality. The proof is as follows.

First, the constraint $\|\Delta_\Phi\|_2 \le \delta_\Phi$ is transformed into $\Delta_\Phi^{\mathrm{T}} \Delta_\Phi \preceq \delta_\Phi^2 I$, then $\bar{\Phi} = \Phi + \Delta_\Phi$ is substituted into $\mathbf{U} \preceq \bar{\Phi}^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\bar{\Phi}$, and inequality (A.2) is received

$$
\begin{aligned}
\mathbf{U} \preceq & \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Phi + \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Delta_\Phi \\
& + \Delta_\Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Phi + \Delta_\Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Delta_\Phi, \\
& \Delta_\Phi^{\mathrm{T}} \Delta_\Phi \preceq \delta_\Phi^2 I.
\end{aligned}
\tag{A.2}
$$

Theorem 3.3 in [15] shows that the above semi-infinite quadratic matrix inequality holds if and only if the following matrix inequality (A.3) holds.

$$
\begin{bmatrix} \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Phi - \mathbf{U} - \beta I & \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H}) \\ \mathrm{diag}\,(\mathbf{H})\Phi & \mathrm{diag}\,(\mathbf{H}) + \left(\dfrac{\beta}{\delta_\Phi^2}\right)\mathbf{I} \end{bmatrix} \succcurlyeq 0, \quad \beta \ge 0.
\tag{A.3}
$$

Thus, the equation (A.1) can be converted into the following inequality:

$$
\mathrm{tr}\left(\mathbf{U}^{-1}\right) \le \tau\gamma_0 - C,
$$

$$
\begin{bmatrix} \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H})\Phi - \mathbf{U} - \beta I & \Phi^{\mathrm{T}} \, \mathrm{diag}\,(\mathbf{H}) \\ \mathrm{diag}\,(\mathbf{H})\Phi & \mathrm{diag}\,(\mathbf{H}) + \left(\dfrac{\beta}{\delta_\Phi^2}\right)\mathbf{I} \end{bmatrix} \succcurlyeq 0, \quad \beta \ge 0.
\tag{A.4}
$$

B. Transformation of Inequality (12)

Setting $\bar{\mathbf{P}}_L = \mathbf{P}_L + \Delta_{PL}$, $\|\Delta_{PL}\|_\infty < \delta_{PL}$, then the optimization constraint (10) can be expressed as

$$
\sup\,\{(\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot \bar{\mathbf{P}}_L\}_i \le (\mathbf{e_0} + \mathbf{E})_i \quad \|\Delta_{PL}\|_\infty < \delta_{PL}, \quad i = 1, \cdots, M.
\tag{B.1}
$$

The left-hand side of equation (B.1) can be expressed as

$$
\begin{aligned}
&\sup\,\left\{\left((\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot \bar{\mathbf{P}}_L\right)_i\right\} \\
&= (\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1} \odot \mathbf{P}_L)_i + \sup\,\{(\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot (\Delta_{PL})_i\} \quad i = 1, \cdots, M \\
&= (\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1} \odot \mathbf{P}_L)_i + \delta_{PL}(\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1})_i \quad i = 1, \cdots, M.
\end{aligned}
\tag{B.2}
$$

Making

$$
\Pi = \left[\underbrace{\delta_{PL}, \delta_{PL}, \cdots, \delta_{PL}}_{M}\right]^{T},
\tag{B.3}
$$

we can vectorize the scalar representation and get the following inequality:

$$
(\mathrm{diag}\,(\mathbf{s})\Theta\mathbf{1}) \odot (\mathbf{P}_L + \Pi) \le \mathbf{e_0} + \mathbf{E}.
\tag{B.4}
$$

Data Availability

For simulation data, please contact the corresponding author of this manuscript.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] B. Wang, J. Zhu, L. T. Yang, and Y. Mo, "Sensor density for confident information coverage in randomly deployed sensor networks," *IEEE Transactions on Wireless Communications*, vol. 15, no. 5, pp. 3238–3250, 2016.

[2] V. Roy, A. Simonetto, and G. Leus, "Spatio-temporal sensor management for environmental field estimation," *Signal Processing*, vol. 128, pp. 369–381, 2016.

[3] X. Wang, X. Liu, C.-T. Cheng, L. Deng, X. Chen, and F. Xiao, "A joint user scheduling and trajectory planning data collection strategy for the UAV-assisted WSN," *IEEE Communications Letters*, vol. 25, no. 7, pp. 2333–2337, 2021.

[4] M. Hibbard, K. Tuggle, and T. Tanaka, "Point-based value iteration and approximately optimal dynamic sensor selection for linear-Gaussian processes," *IEEE Control Systems Letters*, vol. 5, no. 6, pp. 2192–2197, 2021.

[5] H. Jamali, A. Simonetto, and G. Leus, "Sparsity-aware sensor selection: centralized and distributed algorithms," *IEEE Signal Processing Letters*, vol. 21, no. 2, pp. 217–220, 2014.

[6] J. Ranieri, A. Chebira, and M. Vetterli, "Near-optimal sensor placement for linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 62, no. 5, pp. 1135–1146, 2014.

[7] C. Jiang, Y. C. Soh, and H. Li, "Sensor placement by maximal projection on minimum eigenspace for linear inverse problems," *IEEE Transactions on Signal Processing*, vol. 64, no. 21, pp. 5595–5610, 2016.

[8] C. Jiang, Z. Chen, R. Su, and Y. C. Soh, "Group greedy method for sensor placement," *IEEE Transactions on Signal Processing*, vol. 67, no. 9, pp. 2249–2262, 2019.

[9] J. Luo and P. Pattipati, "Fast optimal and suboptimal any-time algorithms for CDMA multiuser detection based on branch and bound," *IEEE Transactions on Communications*, vol. 52, no. 4, pp. 632–642, 2004.

[10] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 451–462, 2009.

[11] C. Rusu, J. Thompson, and N. M. Robertson, "Sensor scheduling with time, energy, and communication constraints," *IEEE Transactions on Signal Processing*, vol. 66, no. 2, pp. 528–539, 2018.

[12] S. K. Sengijpta, "Fundamentals of statistical signal processing: estimation theory," *Control Engineering Practice*, vol. 37, no. 4, pp. 465-466, 1994.

[13] G. Xu, C. Pang, and X. Duan, "Multi-sensor optimization scheduling for target tracking based on PCRLB and a novel intercept probability factor," *Electronics*, vol. 8, no. 2, pp. 140–162, 2019.

[14] M. Grant, S. Boyd, and Y. Ye, "CVX: Matlab software for disciplined convex programming," in *Book Series Nonconvex Optimization and Its Applications*, pp. 155–210, Springer, 2006.

[15] A. Mutapcic and S. Boyd, "Cutting-set methods for robust convex optimization with pessimizing oracles," *Optimization Methods and Software*, vol. 24, no. 3, pp. 381–406, 2009.

Research Article

Multisensor Management Method for Ground Moving Target Tracking Based on Doppler Blind Zone Information

Yunpu Zhang [iD],[1] **Ganlin Shan** [iD],[1] **Rui Zhang** [iD],[1] **and Xiusheng Duan** [iD][2]

[1]*Shijiazhuang Campus, Army Engineering University, Shijiazhuang 050003, China*
[2]*Department of Mechanical Engineering, Shijiazhuang Tiedao University, Shijiazhuang 050003, China*

Correspondence should be addressed to Yunpu Zhang; zypsensor@163.com

Tracking ground moving target with sensors proves to be a challenge due to the uncertainty of target motion area, the existence of Doppler blind zone (DBZ), and the complex terrain. In this paper, a multisensor management method based on DBZ information is presented, in which the available sensors are selected to obtain the best operational revenues for ground moving target tracking. First, the ground target motion model is established considering the off-road/on-road state based on road topology information. Second, the sensor measurement model is given combined with the DBZ information, and a decorrelation method of measurement noise is proposed. Third, a target state estimation algorithm is derived using particle filter, in which the DBZ information is regarded as prior information. Then, combined with the variable structure interacting multiple model method, an estimation algorithm for tracking maneuvering target is proposed. Furthermore, an optimization model of nonmyopic sensor management is constructed to obtain the best sensor management scheme. Finally, the advancement and effectiveness of the proposed management method are verified in the simulations.

1. Introduction

With the development of sensing technology and information fusion technology, multisensor systems have been widely used in the military field [1–5]. How to determine a reasonable and effective sensor management method to obtain the best operational revenues has become a research hotspot for many scholars.

In terms of decision-making methods in sensor management, there are two kinds of methods, including the myopic sensor management method [6] and the nonmyopic sensor management method [7]. The myopic method decides the management scheme based on predicting the one-step revenue in the future. On the contrast, the nonmyopic method decides the management scheme based on predicting the multistep revenues, which can obtain a better management performance than the myopic method, but with a large computing time.

At present, scholars usually focus their research on the sensor management method based on the optimization indicator, whose connotation is to set an objective function closely related to the optimization indicator to maximize the desired operational revenues [8, 9]. According to the selected optimization indicator, the sensor management methods can be divided into three main categories: information-indicator-based management methods, risk-indicator-based management methods, and task-indicator-based management methods. The information-indicator-based methods focus on managing sensors to maximize the information gain, thus reducing the uncertainty in the observation. Commonly used information indicators are the Shannon entropy [10, 11], the Kullback-Leibler divergence [12], and the Rényi information divergence [13, 14]. However, the disadvantage of the methods is that the meaning of the used information indicators is too abstract to describe their concrete physical meaning, which may make it difficult for commanders to understand their connotation accurately. The risk-indicator-based management methods are mainly used in scenarios where sensor resources are scarce. The methods consider that the sensor management is performed to reduce the losses due to measurement uncertainty, rather than to maximize measurement performance. Commonly

used risk indicators include the threat assessment risk [15], the target loss risk [16, 17], and the sensor radiation risk [18–20]. The task-indicator-based management methods focus on the combat task, and the relevant indicators are directly related to the tasks. Unlike the risk-indicator-based method, the methods are applicable when the sensor resources are redundant. Typical indicators are the covariance matrix of target state [21, 22], the target detection probability [23], and the posterior Cramér-Rao lower bound (PCRLB) [24, 25]. Among the three kinds of sensor management methods mentioned above, methods for target tracking have been most studied, and the representative work is shown in [19, 25, 26]. In [19], a nonmyopic sensor management method is applied to track aerial targets for the tradeoff between the tracking accuracy and the radiation risk. Mohammadi and Asif [25] present a dynamic sensor scheduling for tracking problems in distributed sensor networks, in which the PCRLB is used to quantify the tracking accuracy. Gostar et al. [26] present a constrained sensor management method for the multitarget tracking based on information divergence, and the labeled multi-Bernoulli filter was proposed for estimating the target state.

However, most of the existing studies about sensor management methods mainly focus on aerial targets and neglect ground targets. In the actual battlefield, there is inevitably a need for tracking ground targets, whose motions are uncertain and often subject to the terrain information [27–29]. In ground target tracking, the most widely used terrain information, is the road information, and the representative work is shown in [30–35]. In [30], an on-road target tracking is proposed, in which the linear roads are mapped to the ground coordinate system and the road network is established as a one-dimensional linear mode. Koch et al. [31] propose a road map extraction method to reduce the error of the road map. In [32], the road network is modeled as constant curvature segments, and the nonlinear road-constrained Kalman filtering is used in target state estimation. In [33], the widths of road segments are considered in the state model of road target which influences the transition of the target state. Zheng and Gao [34] introduce the Gaussian mixture probability hypothesis density filter in the ground target tracking and proposed a multitarget tracking method in clutter combined with the road constraints. In [35], a ground moving target indication (GMTI) tracking problem is presented, and a variable structure interacting multiple model (VSIMM) particle filter was applied in the switching problem of road segments.

Meanwhile, the ground target detection sensors, such as the GMTI radar, mostly used the pulsed Doppler techniques to reduce the effect of clutters [36]. However, the Doppler blind zone (DBZ) problem has become a great challenge in target tracking. The low-Doppler targets, whose Doppler magnitude falls (i.e., the radial velocity of the target) below the minimum detectable velocity (MDV), cannot be detected by the sensor [37]. This results in a significant decrease in tracking accuracy. For the target tracking in the presence of the DBZ, most previous works focus on improving the filter and data fusion algorithms [38–40], but few works utilize sensor management to solve it.

To solve the problem mentioned above, we propose a multisensor management method for ground moving target tracking based on the DBZ information. The study focuses on the following points.

First, a more realistic ground moving target tracking scenario is considered in which the target can move in different motion areas and its movement is constrained by road topology information. That is to say, the target motion state is divided into two categories: off-road and on-road.

Second, a sensor measurement model under measurement uncertainty is presented considering the DBZ information. Meanwhile, Lei and Han in [41] indicate that the measurement noise of the range and the radial velocity are statistically correlated. To solve this problem, we propose a decorrelation method of measurement noise.

Third, combined with the particle filter (PF) algorithm, a target state estimation algorithm based on the DBZ information is proposed, in which the DBZ is regarded as prior information. Then, a VSIMM-PF-DBZ algorithm is proposed for tracking the maneuvering target combining the VSIMM method.

Finally, we introduce the PCRLB to quantify the tracking accuracy in the future and establish a nonmyopic sensor management optimization model to obtain the optimal management scheme.

The research framework of this paper is as follows. Section 2 describes the sensor management problem. The ground target motion model and the sensor measurement model are established in Section 3 and Section 4, respectively. In Section 5, the target state estimation algorithm based on the DBZ information is proposed. In Section 6, the VSIMM-PF-DBZ algorithm is presented. The sensor management optimization model is given in Section 7. Section 8 makes some simulations to illustrate the effectiveness and advancement of the proposed management method. Finally, conclusions are given in Section 9.

2. Problem Description

Sensors are Doppler radars in this paper. The target tracking scene is shown in Figure 1. Assume that there are N sensors to track a ground moving target, which can move on the road or off-road. The data processing center unifies the measurements obtained by the sensors, searches the sensor management scheme with the best revenues, and sends the corresponding commands to control the working of the sensor system. Note that we use the nonmyopic sensor management method in which the management scheme is decided based on the cumulative revenues over a time horizon in the future [7, 42].

At time k, the sensor management scheme is denoted as $A_k = (a_k^n)_{N \times 1}$, where a_k^n indicates the working state of sensor n. If sensor n is activated to track the target at time k, then $a_k^n = 1$; otherwise, $a_k^n = 0$. Then, the management scheme sequence over the H − step time horizon is denoted as $A_{k:k+H-1} = [A_k, A_{k+1}, \cdots, A_{k+H-1}]$.

For the convenience of presentation, we set the following two constraints related to $\mathbf{A}_{k:k+H-1}$:
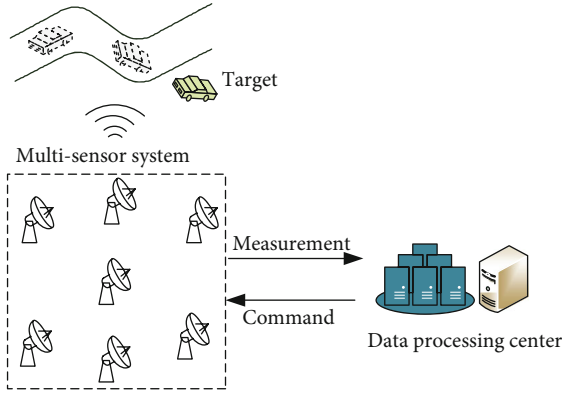
FIGURE 1: Target tracking scene.

(1) One target must be tracked by only one sensor at each time

(2) Due to the sensors cannot be switched frequently in practice, we consider the minimum dwell time $t_{\min}$ in sensor selecting. Before the working sensor can be switched, its continuous working time must exceed $t_{\min}$

The above constraints can be written as

$$\begin{cases} \sum_{n=1}^{N} a_{k+h-1}^{n} = 1, 1 \leq h \leq H \\ t(A_{k+h-1}) \geq t_{\min} \end{cases}, \qquad (1)$$

where $t(A_{k+h-1})$ represents the continuous working time of the switched sensor at time $k + h - 1$.

3. Ground Target Motion Model

The ground target state at time k is $X_k = [x_k, \dot{x}_k, y_k, \dot{y}_k]^{\mathrm{T}}$, where x_k and y_k are the position coordinates in the real-world coordinate system and $\dot{x}_k$ and $\dot{y}_k$ are the corresponding velocities. When a target moves on the road, its motion state is mainly affected by the road topology [43]. For the off-road target, its motion is relatively free. Therefore, different state models are required to describe the characteristics of the target motion in different areas.

3.1. Off-Road State Model. The off-road state model can be described as [19]

$$X_k = F_i^{\text{off-road}} X_{k-1} + G_i^{\text{off-road}} \varepsilon_k^{\text{off-road}}, \qquad (2)$$

where $F_i^{\text{off-road}}$ is the state transition matrix of motion model i, $G_i^{\text{off-road}}$ is the corresponding process noise gain matrix, $\varepsilon_k^{\text{off-road}}$ is the zero means Gaussian process noise with covariance matrix $Q_k^{\text{off-road}} = \text{diag}\,(\sigma_x, \sigma_y)$, and τ is the sampling interval. In this paper, two typical motion models are considered, including the constant velocity (CV) model and the constant turn (CT) model. Thus, the corresponding state transition matrixes and process noise gain matrices are

as follows [24].

$$F_{\text{CV}}^{\text{off-road}} = \begin{bmatrix} 1 & \tau & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & \tau \\ 0 & 0 & 0 & 1 \end{bmatrix}, \qquad (3)$$

$$F_{CT}^{off-road} = \begin{bmatrix} 1 & \sin\varphi\pi/\varphi & 0 & -(1-\cos\varphi\pi)/\varphi \\ 0 & \cos\varphi\pi & 0 & -\sin\varphi\pi \\ 0 & (1-\cos\varphi\pi)/\varphi & 1 & \sin\varphi\pi/\varphi \\ 0 & \sin\varphi\pi & 0 & \cos\varphi\pi \end{bmatrix}, \qquad (4)$$

$$G_{\text{CV}}^{\text{off-road}} = G_{\text{CT}}^{\text{off-road}} = \begin{bmatrix} \tau^2/2 & 0 \\ \tau & 0 \\ 0 & \tau^2/2 \\ 0 & \tau \end{bmatrix}, \qquad (5)$$

where φ is the turn rate of the target.

3.2. On-Road State Model. The information of road segments can be collected from geographic information systems (GIS) [43]. Then, the mathematical model of the road network is established by using relevant information to represent the road network as a connection of many road segments, as shown in Figure 2. When the target is moving continuously on the road, it can be considered as moving along the road centerline without deviating largely normal to it.

When the target is moving along the centerline of road segment l whose start point is $(x_l^{\text{start}}, y_l^{\text{start}})$ and end point is $(x_l^{\text{end}}, y_l^{\text{end}})$, its state constraint can be described as

$$\begin{cases} a_l x_k + b_l y_k + c_l = 0, \\ \left\langle [\dot{x}_k, \dot{y}_k]^{\mathrm{T}}, \vec{\varsigma} \right\rangle = 0, \end{cases} \qquad (6)$$

with

$$\begin{cases} a_l = y_l^{\text{start}} - y_l^{\text{end}}, \\ b_l = x_l^{\text{end}} - x_l^{\text{start}}, \\ c_l = \left(y_l^{\text{end}} - y_l^{\text{start}}\right) x_l^{\text{start}} - \left(x_l^{\text{end}} - x_l^{\text{start}}\right) y_l^{\text{start}}, \end{cases} \qquad (7)$$

where $\left\langle [\dot{x}_k, \dot{y}_k]^{\mathrm{T}}, \vec{\varsigma} \right\rangle$ represents the angle between two vectors, $\vec{\varsigma}$ is the direction vector of the road segment direction, a_l, b_l, and c_l are the coefficients of the centerline equation.

Equation (6) considers the state constraint when the target moves along the centerline, but the target may slightly deviate to the centerline because of the presence of the process noise. Combined with equation (6), the target state
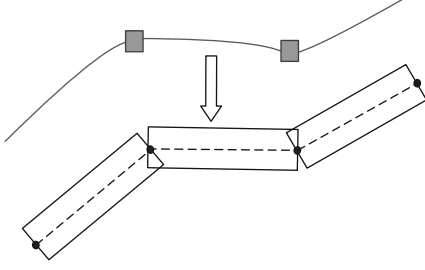
FIGURE 2: Road network modeling schematic.

constraint can be expressed as

$$
\begin{cases}
d[(x_k, y_k)|a_l x + b_l y + c_l = 0] \leq \Delta d \\
\left\langle [\dot{x}_k, \dot{y}_k]^{\mathrm{T}}, \vec{\varsigma} \right\rangle \leq \Delta v
\end{cases}, \tag{8}
$$

where $d[(x_k, y_k)|a_l x + b_l y + c_l = 0]$ represents the distance from point (x_k, y_k) to line $a_l x + b_l y + c_l = 0$ and Δd and Δv are the deviation threshold of the distance and velocity angle, respectively.

While satisfying the above state constraint, the on-road state model can be described as

$$
X_k = F_i^{\text{on-road}} X_{k-1} + G_i^{\text{on-road}} \boldsymbol{\varepsilon}_k^{\text{on-road}}. \tag{9}
$$

In this paper, we consider that the on-road target mentioned only moves in CV model. Therefore, $F_i^{\text{on-road}} = F_{\text{CV}}^{\text{off-road}}$, and $G_i^{\text{on-road}} = G_{\text{CV}}^{\text{off-road}}$.

We denote the variances of the process noise along the road and orthogonal to the road as $\sigma_\parallel^2$ and $\sigma_\perp^2$ ($\sigma_\perp^2 \ll \sigma_\parallel^2$), respectively. The covariance matrix $\mathbf{Q}_k^{\text{on-road}}$ after projecting it to the X-Y coordinate system can be written as [33]

$$
Q_k^{\text{on-road}} = \begin{bmatrix} \cos\theta_l & -\sin\theta_l \\ \sin\theta_l & \cos\theta_l \end{bmatrix} \begin{bmatrix} \sigma_\parallel^2 & 0 \\ 0 & \sigma_\perp^2 \end{bmatrix} \begin{bmatrix} \cos\theta_l & -\sin\theta_l \\ \sin\theta_l & \cos\theta_l \end{bmatrix}^{\mathrm{T}}, \tag{10}
$$

where θ_l represents the orientation angle of road segment l.

Note that the state transition of the on-road target is also determined by the road segment on which they are located, while the state transition of the off-road target is mainly determined by its motion model (CV or CT).

3.3. Motion Area Transitions.

Obviously, the motion areas of the target can be switched between on-road and off-road. In this paper, parameter m_k is used to indicate the area in which the target is moving at time k, where $m_k = 0$ represents the target moving on the off-road area and $m_k = 1$ represents the target moving on the road. The transition process of motion area can be approximated as a Markov process, and the cor-

responding transition matrix can be stated as

$$
T_k = \begin{bmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{bmatrix} = \begin{bmatrix} p(m_k = 0|m_{k-1} = 0) & p(m_k = 1|m_{k-1} = 0) \\ p(m_k = 0|m_{k-1} = 1) & p(m_k = 1|m_{k-1} = 1) \end{bmatrix}, \tag{11}
$$

with

$$
\begin{cases}
p(m_k = 0|m_{k-1} = 0) = 1 - \exp\left(-\vartheta \cdot d_k^{\text{entry}}\right), \\
p(m_k = 1|m_{k-1} = 0) = \exp\left(-\vartheta \cdot d_k^{\text{entry}}\right), \\
p(m_k = 0|m_{k-1} = 1) = \exp\left(-\vartheta \cdot d_k^{\text{exit}}\right), \\
p(m_k = 1|m_{k-1} = 1) = 1 - \exp\left(-\vartheta \cdot d_k^{\text{exit}}\right).
\end{cases} \tag{12}
$$

Here, d_k^{entry} and d_k^{exit} represent the distance to the nearest entry point and nearest exit point of the road network, respectively. ϑ is a probability parameter (in this paper, $\vartheta = 0.04$).

4. Sensor Measurement Model

4.1. Sensor Measurement Equation.

The Doppler sensor can obtain the distance, azimuth, and radial velocity information of the target, and the corresponding measurement equation can be expressed as [19]

$$
Z_k^n = H^n(X_k) + \mathbf{v}^n = \begin{bmatrix} r_k^n \\ \varphi_k^n \\ \dot{r}_k^n \end{bmatrix} + \begin{bmatrix} v_r^n \\ v_\varphi^n \\ v_r^n \end{bmatrix}, \tag{13}
$$

with

$$
\begin{cases}
r_k^n = \sqrt{(x_k - x_s^n)^2 + (y_k - y_s^n)^2}, \\
\varphi_k^n = \arctan\left(\dfrac{y_k - y_s^n}{x_k - x_s^n}\right), \\
\dot{r}_k^n = \dfrac{\dot{x}_k(x_k - x_s^n) + \dot{y}_k(y_k - y_s^n)}{\sqrt{(x_k - x_s^n)^2 + (y_k - y_s^n)^2}},
\end{cases} \tag{14}
$$

where the measurement information $[r_k^n \quad \varphi_k^n \quad \dot{r}_k^n]^{\mathrm{T}}$ represent the range, azimuth, and radial velocity of the target at time k and the noise $\mathbf{v}^n = [v_r^n \quad v_\varphi^n \quad v_r^n]^{\mathrm{T}}$ represent the Gaussian measurement noise with zero means. x_s^n and y_s^n are the position coordinates of sensor n.

If the target is not detected successfully, the sensor cannot obtain the measurement information. When the DBZ is not considered, the detection probability of sensor n can

be written as [44]

$$\tilde{p}_d^n(X_k) = \begin{cases} \left(p_f^n\right)^{1/(1+SNR)}, & \text{if the target is inside line-of-sight region} \\ 0, & \text{otherwise} \end{cases}, \tag{15}$$

with

$$SNR = SNR_{\min}^n \left(\frac{R_{\max}^n}{r_k^n}\right)^4, \tag{16}$$

where SNR is the signal to noise ratio and p_f^n, $SNR_{\min}^n$, and $R_{\max}^n$ represent the false probability, the minimum detectable SNR, and the maximum detection distance of sensor n, respectively.

When the DBZ is considered, the target with radial velocity below than MDV will not be detected, which is an important priori information for sensor management. The corresponding detection probability can be calculated by [37]

$$p_d^n(X_k) = \begin{cases} \tilde{p}_d^n(X_k)\left\{1 - \exp\left[-\log 2 \left(\frac{n_c(X_k)}{V_{MDV}^n}\right)^2\right]\right\}, & \text{if target is inside line-of-sight region} \\ & \text{and } |\dot{r}_k| > V_{MDV}^n \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where V_{MDV}^n is the MDV of sensor n and $n_c(X_k)$ is called the clutter notch function (see [37] for details).

Combined with the detection probability, the measurement equation can be written as

$$Z_k^n = \eta H^n(X_k) + \mathbf{v}^n, \tag{18}$$

where η represents a random number, which is taken as 0 or 1 according to the detection probability $p_d^n(\mathbf{X}_k)$, that is

$$p(\eta) = \begin{cases} p_d^n(X_k), & \eta = 1 \\ 1 - p_d^n(X_k), & \eta = 0 \end{cases}. \tag{19}$$

4.2. Decorrelation Method of the Measurement Noise. According to [41], the range measurement noise v_r^n and the radial velocity measurement noise $v_{\dot{r}}^n$ are statistically correlated, which cannot be ignored in the process of target tracking. Hence, we propose a decorrelation method of measurement noise to improve the accuracy of target tracking in this paper.

We define $\sigma^2(v_r^n)$, $\sigma^2(v_\varphi^n)$, and $\sigma^2(v_{\dot{r}}^n)$ as the variances of v_φ^n, v_φ^n, and v_r^n, respectively. The correlation coefficient of $\sigma^2(v_r^n)$ and $\sigma^2(v_{\dot{r}}^n)$ is denoted as ϖ. Then, the covariance matrix of measurement noise can be written as

$$R^n = \begin{bmatrix} \sigma^2(v_r^n) & 0 & \varpi\sigma(v_r^n)\sigma(v_{\dot{r}}^n) \\ 0 & \sigma^2(v_\varphi^n) & 0 \\ \varpi\sigma(v_r^n)\sigma(v_{\dot{r}}^n) & 0 & \sigma^2(v_{\dot{r}}^n) \end{bmatrix}. \tag{20}$$

Rewrite equation (20) as

$$R^n = \begin{bmatrix} R_{r,\varphi}^n & \left(R_{r,\dot{r}}^n\right)^T \\ R_{r,\dot{r}}^n & R_{\dot{r},\dot{r}}^n \end{bmatrix}, \tag{21}$$

with

$$\begin{cases} R_{r,\varphi}^n = \text{diag}\left[\sigma^2(v_r^n), \sigma^2\left(v_\varphi^n\right)\right], \\ R_{r,\dot{r}}^n = [\varpi\sigma(v_r^n)\sigma(v_{\dot{r}}^n), 0], \\ R_{\dot{r},\dot{r}}^n = \sigma^2(v_{\dot{r}}^n). \end{cases} \tag{22}$$

To eliminate the correlation of the elements in R^n, we use the Cholesky decomposition method and construct the decomposition matrix

$$B^n = \begin{bmatrix} I_{2\times 2} & 0 \\ L^n & 1 \end{bmatrix}, \tag{23}$$

with

$$L^n = -R_{r,\dot{r}}^n \left(R_{r,\varphi}^n\right)^{-1} = [-\varpi\sigma(v_{\dot{r}}^n)/\sigma(v_r^n), 0], \tag{24}$$

where $I_{2\times 2}$ is an identity matrix.

Multiplying both sides of equation (18) simultaneously left by B^n, the decorrelated measurement equation is obtained as follows

$$\begin{cases} \tilde{Z}_k^n = \eta \begin{bmatrix} r_k^n \\ \varphi_k^n \end{bmatrix} + \begin{bmatrix} v_r^n \\ v_\varphi^n \end{bmatrix}, \\ \tilde{\psi}_k^n = \eta\psi_k^n + v_\psi^n, \end{cases} \tag{25}$$

where $\tilde{\psi}_k^n$ represents the pseudomeasurement of the radial velocity and ψ_k and v_ψ^n are the measurement value and noise of $\tilde{\psi}_k^n$, respectively, which can be calculated by

$$\psi_k^n = \frac{-\varpi\sigma(v_{\dot{r}}^n)}{\sigma(v_r^n)} r_k + \dot{r}_k, \tag{26}$$

$$v_\psi^n = \frac{-\varpi\sigma(v_{\dot{r}}^n)}{\sigma(v_r^n)} v_r^n + v_{\dot{r}}^n. \tag{27}$$

Obviously, v_ψ^n is the Gaussian noise with zero means, whose variance can be expressed as

$$\sigma^2\left(v_\psi^n\right) = \text{cov}\left(v_\psi^n, v_\psi^n\right) = \left(1 - \varpi^2\right)\sigma^2(v_{\dot{r}}^n). \tag{28}$$

Therefore, the new measurement equation can be

written as

$$
\widehat{Z}_k^n = \eta \widehat{H}^n(X_k) + \widehat{\mathbf{v}}^n = \eta \begin{bmatrix} r_k^n \\ \varphi_k^n \\ \psi_k^n \end{bmatrix} + \begin{bmatrix} v_r^n \\ v_\varphi^n \\ v_\psi^n \end{bmatrix}. \tag{29}
$$

The corresponding covariance matrix of the measurement noise is

$$
\widehat{R}^n = \begin{bmatrix} \sigma^2(v_r^n) & 0 & 0 \\ 0 & \sigma^2\left(v_\varphi^n\right) & 0 \\ 0 & 0 & (1 - \varpi^2)\sigma^2(v_r^n) \end{bmatrix}. \tag{30}
$$

5. Target State Estimation Algorithm Based on the DBZ Information

In the process of sensor management, estimating the target state at each moment is a prerequisite for system decision-making. When the DBZ is considered, the sensors will lose the measurements in some cases, and the target state only can be obtained by prediction based on the recurrence of the target motion models. At this time, the tracking accuracy will be greatly reduced, which is not conducive to the stable tracking of the target.

It is known that the target state in the DBZ satisfies a certain constraint relationship, that is, the radial velocity of the target is below than the MDV of the sensor, which can be used to improve the tracking performance as prior information when the sensor cannot obtain the measurements. Therefore, we propose a particle filter algorithm based on the DBZ information (PF-DBZ) for target state estimation in this paper.

For the convenience of expression, the multimodel and motion area transitions of the target are not considered in this section.

5.1. Process of the PF-DBZ. The generated particles are divided into two categories: one is the particles outside the blind zone called unconstrained particles with a number of M_{UN}, denoted as $\{\mathbf{X}_{\mathrm{UN},k}^i : i = 1, 2, \cdots, M_{\mathrm{UN}}\}$; another is the particles in the DBZ with a number of M_{DBZ}, called DBZ particles, denoted as $\{\mathbf{X}_{\mathrm{DBZ},k-1}^i : i = 1, 2, \cdots, M_{\mathrm{DBZ}}\}$, which are constrained by the DBZ information. At time k, the occurrence probability of the two kinds of particles is $p_{\mathrm{UN},k}$ and $p_{\mathrm{DBZ},k}$, respectively. Then, the processes of PF-DBZ are as follows.

Step 1. State prediction.

For the unconstrained particles, the prediction state $\tilde{X}_{\mathrm{UN},k}^i (i = 1, 2, \cdots, M_{\mathrm{UN}})$ can be calculated based on the state model mentioned in Section 3:

$$
\tilde{X}_{\mathrm{UN},k}^i = F X_{\mathrm{UN},k-1}^i + G_k Q_k (G_k)^{\mathrm{T}}. \tag{31}
$$

For the DBZ particles, if $p_{\mathrm{DBZ},k-1} > 0$, the prediction state $\tilde{X}_{\mathrm{DBZ},k}^i (i = 1, 2, \cdots, M_{\mathrm{DBZ}})$ can also be obtained by equation (31), but the state must meet the following DBZ information constraint.

$$
\frac{\left(\dot{\tilde{x}}_{\mathrm{DBZ},k}^i - \dot{x}_s^n\right)\left(\tilde{x}_{\mathrm{DBZ},k}^i - x_s^n\right) + \left(\dot{\tilde{y}}_{\mathrm{DBZ},k}^i - \dot{y}_s^n\right)\left(\tilde{y}_{\mathrm{DBZ},k}^i - y_s^n\right)}{\sqrt{\left(\tilde{x}_{\mathrm{DBZ},k}^i - x_s^n\right)^2 + \left(\tilde{y}_{\mathrm{DBZ},k}^i - y_s^n\right)^2}} \leq V_{\mathrm{MDV}}^n. \tag{32}
$$

Step 2. Measurement update.

At time k, whether sensor n can obtain the measurement is an uncertain event. Therefore, different cases will be discussed as follows.

(1) Calculate the weight of the particles. The weights of the unconstrained particles $\tilde{X}_{\mathrm{UN},k}^i$ can be obtained according to the likelihood function corresponding to the sensor measurement equation, that is, $\omega_{\mathrm{UN},k}^i \sim p(\widehat{Z}_k^n | \tilde{X}_{\mathrm{UN},k}^i)$. In this paper, the likelihood function is

$$
p\left(\widehat{Z}_k^n \middle| \tilde{X}_{\mathrm{UN},k}^i\right) = \frac{1}{\sqrt{2\pi|\widehat{R}^n|}} \exp\left[-\frac{1}{2}\left[\widehat{Z}_k^n - \widehat{H}\left(\tilde{X}_{\mathrm{UN},k}^i\right)\right]^{\mathrm{T}}\left(\widehat{R}^n\right)^{-1}\left[\widehat{Z}_k^n - \widehat{H}\left(\tilde{X}_{\mathrm{UN},k}^i\right)\right]\right] \tag{33}
$$

(2) Normalize the weights:

$$
\omega_{\mathrm{UN},k}^i = \frac{\omega_{\mathrm{UN},k}^i}{\sum_{i=1}^{M_{\mathrm{UN}}} \omega_{\mathrm{UN},k}^i} \tag{34}
$$

(3) Update the occurrence probability of two kinds of particles. Set $p_{\mathrm{UN},k} = 1$ and $p_{\mathrm{DBZ},k} = 0$

(1) Calculate the weight of the particles. Set $\omega_{\mathrm{UN},k}^i = \omega_{\mathrm{UN},k-1}^i$ and $\omega_{\mathrm{DBZ},k}^i = 1/M_{\mathrm{DBZ}}$

(2) Update the occurrence probability of two kinds of particles. Set $p_{\mathrm{UN},k} = 1 - \tilde{p}_{\mathrm{d}}^n(X_k)$ and $p_{\mathrm{DBZ},k} = \tilde{p}_{\mathrm{d}}^n(X_k)$

(1) Calculate the weight of the particles. Set $\omega_{\mathrm{UN},k}^i = \omega_{\mathrm{UN},k-1}^i$ and $\omega_{\mathrm{DBZ},k}^i = 1/M_{\mathrm{DBZ}}$

(2) Update the occurrence probability of two kinds of particles. Set $p_{\mathrm{UN},k} = [1 - \tilde{p}_{\mathrm{d}}^n(X_k)]p_{\mathrm{UN},k-1}$ and $p_{\mathrm{DBZ},k} = 1 - p_{\mathrm{UN},k}$

Case 1. Measurement Z_k^n is existing.

Case 2. Z_k^n is not existing and $p_{\text{DBZ},k-1} = 0$.

Case 3. Z_k^n is not existing and $p_{\text{DBZ},k-1} > 0$ (continuous missing detection).

$$
\begin{cases}
\widehat{X}_k = p_{\text{UN},k} \displaystyle\sum_{i=1}^{M_{\text{UN}}} \omega_{\text{UN},k}^i \tilde{X}_{\text{UN},k}^i ++ p_{\text{DBZ},k} \sum_{i=1}^{M_{\text{DBZ}}} \omega_{\text{DBZ},k}^i \tilde{X}_{\text{DBZ},k}^i, \\[3mm]
P_k = p_{\text{UN},k} \displaystyle\sum_{i=1}^{M_{\text{UN}}} \omega_{\text{UN},k}^i \left(\widehat{X}_{\text{UN},k} - \tilde{X}_{\text{UN},k}^i\right)\left(X\wedge_{\text{UN},k} - \tilde{X}_{\text{UN},k}^i\right)^{\text{T}} + p_{\text{DBZ},k} \sum_{i=1}^{M_{\text{DBZ}}} \omega_{\text{DBZ},k}^i \left(\widehat{X}_{\text{DBZ},k} - \tilde{X}_{\text{DBZ},k}^i\right)\left(X\wedge_{\text{DBZ},k} - \tilde{X}_{\text{DBZ},k}^i\right)^{\text{T}},
\end{cases}
$$

$$(35)$$

with

$$
\begin{cases}
\widehat{X}_{\text{UN},k} = \displaystyle\sum_{i=1}^{M_{\text{UN}}} \omega_{\text{UN},k}^i \tilde{X}_{\text{UN},k}^i, \\[3mm]
\widehat{X}_{\text{DBZ},k} = \displaystyle\sum_{i=1}^{M_{\text{DBZ}}} \omega_{\text{DBZ},k}^i \tilde{X}_{\text{DBZ},k}^i.
\end{cases}
\qquad (36)
$$

5.2. Correction Method of Target State. According to Section 5.1, the prediction state of the DBZ particles must meet the Doppler information constraint in the process of state prediction. For the particles that do not meet the constraint, the DBZ information can be used to correct their state to improve the accuracy of target state prediction.

For the prediction state $\tilde{X}_{\text{DBZ},k}^i = [\tilde{x}_{\text{DBZ},k}^i, \dot{\tilde{x}}_{\text{DBZ},k}^i, \tilde{y}_{\text{DBZ},k}^i, \dot{\tilde{y}}_{\text{DBZ},k}^i]$ of DBZ particles, when its radial velocity $\dot{\tilde{r}}_k^n > V_{\text{MDV}}^n$, the correction method is expressed as

$$
\begin{cases}
\dot{\tilde{x}}_{\text{DBZ},k}^i = \dot{\tilde{x}}_{\text{DBZ},k}^i - \left(\dot{\tilde{r}}_k^n - V_{\text{MDV}}^n\right) \sin\left(\tilde{\varphi}_k^n\right), \\[2mm]
\dot{\tilde{y}}_{\text{DBZ},k}^i = \dot{\tilde{y}}_{\text{DBZ},k}^i - \left(\dot{\tilde{r}}_k^n - V_{\text{MDV}}^n\right) \cos\left(\tilde{\varphi}_k^n\right),
\end{cases}
\qquad (37)
$$

where $\tilde{\varphi}_k^n$ is the prediction value of the azimuth.

When $\dot{\tilde{r}}_k^n < -V_{\text{MDV}}^n$, the correction method is expressed as

$$
\begin{cases}
\dot{\tilde{x}}_{\text{DBZ},k}^i = \dot{\tilde{x}}_{\text{DBZ},k}^i - \left(\dot{\tilde{r}}_k^n + V_{\text{MDV}}^n\right) \sin\left(\tilde{\varphi}_k^n\right), \\[2mm]
\dot{\tilde{y}}_{\text{DBZ},k}^i = \dot{\tilde{y}}_{\text{DBZ},k}^i - \left(\dot{\tilde{r}}_k^n + V_{\text{MDV}}^n\right) \cos\left(\tilde{\varphi}_k^n\right).
\end{cases}
\qquad (38)
$$

5.3. Judgment Method of Measurement Loss Causes. In ground moving target tracking, the occlusion of obstacles may also cause continuous missed detection. Therefore, it is necessary to distinguish whether the target is occluded by obstacles or enters the DBZ. According to the motion characteristics of the ground target, the radial velocity has

Step 3. State estimation.

The state estimation and covariance matrix after weighted average are given by

a general tendency to decrease gradually before the target enters the DBZ. Therefore, a radial velocity sliding window is used to judge measurement loss causes in this paper.

The length of the sliding window is set to 5, and the historical radial velocity $\{\dot{r}_{k-1}^n, \dot{r}_{k-2}^n, \dot{r}_{k-3}^n, \dot{r}_{k-4}^n, \dot{r}_{k-5}^n\}$ is stored in the sliding window at time k. Then, we define $\nabla\dot{r}_{k-i}^n$ as the radial velocity change rate, which is expressed as

$$
\nabla\dot{r}_{k-i}^n = \left(|\dot{r}_{k-i}^n| - \left|\dot{r}_{k-(i-1)}^n\right|\right)/\tau, \quad i = 1, 2, 3, 4, \qquad (39)
$$

where τ is the sampling interval.

Using the three-fourths judgment rule, when three inequalities in equation (40) satisfy the condition, it is judged that the reason of losing measurement is entering the DBZ. In this case, the PF-DBZ and correction method of target state proposed in this paper can be used. Otherwise, it shows that the target is occluded by obstacles, and the target state only can be obtained by prediction based on the recurrence of the target motion models.

$$
\begin{cases}
\nabla\dot{r}_{k-1}^n = (|\dot{r}_{k-1}^n| - |\dot{r}_{k-2}^n|)/\tau \le 0, \\[2mm]
\nabla\dot{r}_{k-2}^n = (|\dot{r}_{k-2}^n| - |\dot{r}_{k-3}^n|)/\tau \le 0, \\[2mm]
\nabla\dot{r}_{k-3}^n = (|\dot{r}_{k-3}^n| - |\dot{r}_{k-4}^n|)/\tau \le 0, \\[2mm]
\nabla\dot{r}_{k-4}^n = (|\dot{r}_{k-4}^n| - |\dot{r}_{k-5}^n|)/\tau \le 0.
\end{cases}
\qquad (40)
$$

6. VSIMM-PF-DBZ Algorithm

The target state estimation algorithm mentioned in Section 5 is suitable for the target with a single motion model. However, according to Section 2, the motion of the ground target involves the switching of the motion model, the located road segment, and the motion area, which is difficult to describe its motion characteristics by only a single model. Hence, we combine the PF-DBZ algorithm with the VSIMM (see [45] for details) and propose a VSIMM-PF-DBZ algorithm to estimate the state of maneuvering target. In the VSIMM method, the motion model set of the target is variable at
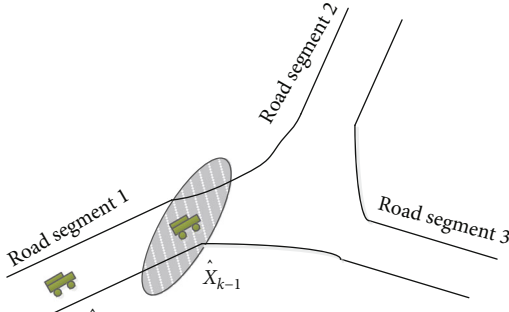
FIGURE 3: Schematic diagram of intersection.

FIGURE 4: Schematic diagram of target tracking scene.

TABLE 1: The parameters of the sensors.

Sensor number	Coordinates	The standard deviation of the range noise	The standard deviation of the azimuth noise	The standard deviations of the radial velocity noise
1	(0, 500) m	40 m	5 mrad	10 m/s
2	(1000, -250) m	20 m	5 mrad	5 m/s
3	(1000, 1500) m	25 m	8 mrad	10 m/s
4	(2550, 500) m	35 m	5 mrad	15 m/s
5	(4000, 0) m	45 m	5 mrad	25 m/s
6	(5500, 1500) m	50 m	7.5 mrad	20 m/s

each time, which can accurately describe the motion state of the multimodel maneuvering target.

At time k, the motion model set is denoted as $J_k = \{ J_k^{\text{off-road}}, J_k^{\text{on-road}} \}$, where $J_k^{\text{off-road}}$ and $J_k^{\text{on-road}}$ represent the model set of off-road and on-road, respectively. In this paper, $J_k^{\text{off-road}}$ is fixed which contains CV and CT models, and $J_k^{\text{on-road}}$ changes in real time according to the target location. Therefore, updating J_k means updating $J_k^{\text{on-road}}$.

When the target moves in the road network, its motion is mainly constrained by the road topology, so the road information can be used to assist the target estimation. Since the road information can be obtained from GIS in advance and generally does not change in target tracking, the motion model set update strategy can be designed based on the road information.

6.1. Motion Model Set Update Strategy. The motion model of the target moving on a certain road segment will generally not change, and only at intersections will the target switch the road segment and its motion model will be changed. Therefore, to update the model set of the next time, it is necessary to determine which road segment the target will be located at the next time and whether the target is close to the intersection.

As seen from Figure 3, the possible location of the target can be obtained according to the target state $\widehat{X}_{k-1}^n$. Then, it can be judged whether the target is close to the intersection, such as the ellipse region in Figure 3. The elliptic region can be expressed as an inequality

$$[x - \widehat{x}_{k-1}, y - \widehat{y}_{k-1}] P_{k-1}^{\text{pos}} \begin{bmatrix} x - \widehat{x}_{k-1} \\ y - \widehat{y}_{k-1} \end{bmatrix} \leq \beta, \quad (41)$$

where P_{k-1}^{pos} is the position component of P_{k-1} and β represents the gate threshold (in this paper, $\beta = 6$).

If intersection i with coordinates $(x_{\text{intersection}}^i, y_{\text{intersection}}^i)$ satisfies inequality (41), it means that the target is close to the intersection, and the model set needs to be updated. The processes of updating $J_{k-1}^{\text{on-road}}$ are as follows.

Step 1. Get the target state $(\widehat{X}_{k-1}, P_{k-1})$ and model set $J_{k-1}^{\text{on-road}}$ at time k.

Step 2. Judge whether the target has closed to the intersection according to equation (41). If the target is close to the intersection, the corresponding motion models of all the links connected to the intersection are added to the model set $J_k^{\text{on-road}}$; otherwise, go to Step 3.

Step 3. Determine the road segment to which the target belongs at time k according to $\widehat{X}_{k-1}^n$. If the target belongs to the road section changes, the motion model of the previous road segment is discarded and the motion model of the new road segment is added in $J_k^{\text{on-road}}$; otherwise, $J_k^{\text{on-road}} = J_{k-1}^{\text{on-road}}$.

6.2. Process of the VSIMM- PF-DBZ Algorithm. Based on the motion model set updating strategy, the VSIMM-PF-DBZ algorithm is proposed by combining the VSIMM with the PF-DBZ algorithm. Due to the uncertainty of the target motion area, we use dual filter algorithm to obtain the target state estimation of off-road state and on-road state and then apply the weighted fusion method according to motion area

transition probability to obtain the overall state estimation. The steps of VSIMM-PF-DBZ are as follows.

Step 1. Initialization.

Obtain the target state $(\widehat{X}_{k-1}, P_{k-1})$, the motion model set J_{k-1}, and the parameter m_{k-1}. Then, according to Section 3.3, calculate the motion area transition matrix T_k, and obtain $p(m_k = 0|m_{k-1})$ and $p(m_k = 1|m_{k-1})$.

Step 2. Update the motion model set.

According to the motion model set update strategy proposed in Section 6.1, obtain the model set $J_k^{\text{on-road}}$ of on-road state.

Step 3. State estimation.

Use the PF-DBZ algorithm in the framework of the VSIMM [45] to predict and update the target states and corresponding covariance matrices of all motion models. Then, the target state estimation of the on-road state and off-road state can be obtained, which are stated as $(\widehat{X}_k^{\text{on-road}}, P_k^{\text{on-road}})$ and $(\widehat{X}_k^{\text{off-road}}, P_k^{\text{off-road}})$, respectively.

Step 4. Overall estimation.

Combined with $(\widehat{X}_k^{\text{on-road}}, P_k^{\text{on-road}})$, $(\widehat{X}_k^{\text{off-road}}, P_k^{\text{off-road}})$, $p(m_k = 0|m_{k-1})$, and $p(m_k = 1|m_{k-1})$, the overall estimation of the target state is calculated as

$$\begin{cases} \widehat{X}_k = p(m_k = 1|m_{k-1})\widehat{X}_k^{\text{on-road}} + p(m_k = 0|m_{k-1})\widehat{X}_k^{\text{off-road}}, \\ P_k = p(m_k = 1|m_{k-1})\left[P_k^{\text{on-road}} + \left(\widehat{X}_k^{\text{on-road}} - \widehat{X}_k\right)\left(X\wedge_k^{\text{on-road}} - X\wedge_k\right)^{\text{T}}\right] + p(m_k = 0|m_{k-1})\left[P_k^{\text{off-road}} + \left(\widehat{X}_k^{\text{off-road}} - \widehat{X}_k\right)\left(X\wedge_k^{\text{off-road}} - X\wedge_k\right)^{\text{T}}\right]. \end{cases}$$

$$(42)$$

7. Sensor Management Optimization Model

The core of sensor management is predicting the revenue corresponding to different decision schemes in the future and selecting the schemes with the best revenue [19]. Therefore, it is important to select an optimization index for quantifying the revenue. The trace of PCRLB can predict the theoretical lower bound of the target state estimation error, which is often used to reflect the sensor tracking performance [24]. Hence, in this paper, the trace of PCRLB is used as the optimization index in the sensor management process.

According to the relevant theory of PCRLB [24], the following inequality exists.

$$E\left[\left(\widehat{X}_k - X_k\right)\left(X\wedge_k - X_k\right)^{\text{T}}\right] \geq \Psi(X\wedge_k)^{-1}, \qquad (43)$$

where $\Psi(X\wedge_k)^{-1}$ represents the PCRLB of $\widehat{X}_k$, which is the inverse of the Fisher information matrix $\Psi(\widehat{X}_k)$.

$\Psi(\widehat{X}_k)$ satisfies the following recurrence relation:

$$\Psi(\widehat{X}_k) = D_{k-1}^{22} - D_{k-1}^{21}\left(\Psi(X\wedge_{k-1}) + D_{k-1}^{11}\right)^{-1}D_{k-1}^{12} + D_k^Z, \quad (44)$$

with

$$\begin{cases} D_{k-1}^{11} = E\left[-\nabla_{\mathbf{X}_{k-1}}^{\mathbf{X}_{k-1}} \log p(X_k | X_{k-1})\right], \\ D_{k-1}^{12} = \left[D_{k-1}^{21}\right]^{\text{T}} = E\left[-\nabla_{\mathbf{X}_{k-1}}^{\mathbf{X}_k} \log p(X_k | X_{k-1})\right], \\ D_{k-1}^{22} = E\left[-\nabla_{\mathbf{X}_k}^{\mathbf{X}_k} \log p(X_k | X_{k-1})\right], \\ D_k^Z = E\left[-\nabla_{\mathbf{X}_k}^{\mathbf{X}_k} \log p(Z_k | X_k)\right], \end{cases}$$

$$(45)$$

where the symbol ∇ represents the second-order derivative and D_k^Z is the Fisher information gain. When the sensor cannot obtain the measurement, there is $D_k^Z = 0$. Furthermore, for the Gaussian system discussed in this paper, there is

$$\begin{cases} D_{k-1}^{11} = (F_k)^{\text{T}}(Q_k)^{-1}F_k, \\ D_{k-1}^{12} = \left[D_{k-1}^{21}\right]^{\text{T}} = -(F_k)^{\text{T}}(Q_k)^{-1}, \\ D_{k-1}^{22} = (Q_k)^{-1}, \\ D_k^Z = p_{\text{d}}^n(X_k)\left[\widehat{h}^n\right]^{\text{T}}\left(\widehat{R}^n\right)^{-1}\widehat{h}^n, \end{cases}$$

$$(46)$$

where $\widehat{h}^n$ is the Jacobian matrix of measurement function $\widehat{H}^n(X_k)$ and F_k and Q_k represent the state transition matrix and covariance matrix of the process noise corresponding to the real target motion model at time k, respectively. Obviously, it is impossible to obtain the real target motion model at time k in the process of decision-making, so the motion model corresponding to the maximum distribution probability at the current time is selected to do prediction [24].

In this paper, the sensor management method is non-myopic, in which the management scheme is decided based on the cumulative revenues over a time horizon in the future. When the decision step is H, combining the sensor management scheme $A_{k:k+H-1}$ in the time domain $[k, k+H-1]$ and the constraints equation (1), the objective

(a) Sensor 1

(b) Sensor 2

(c) Sensor 3

(d) Sensor 4
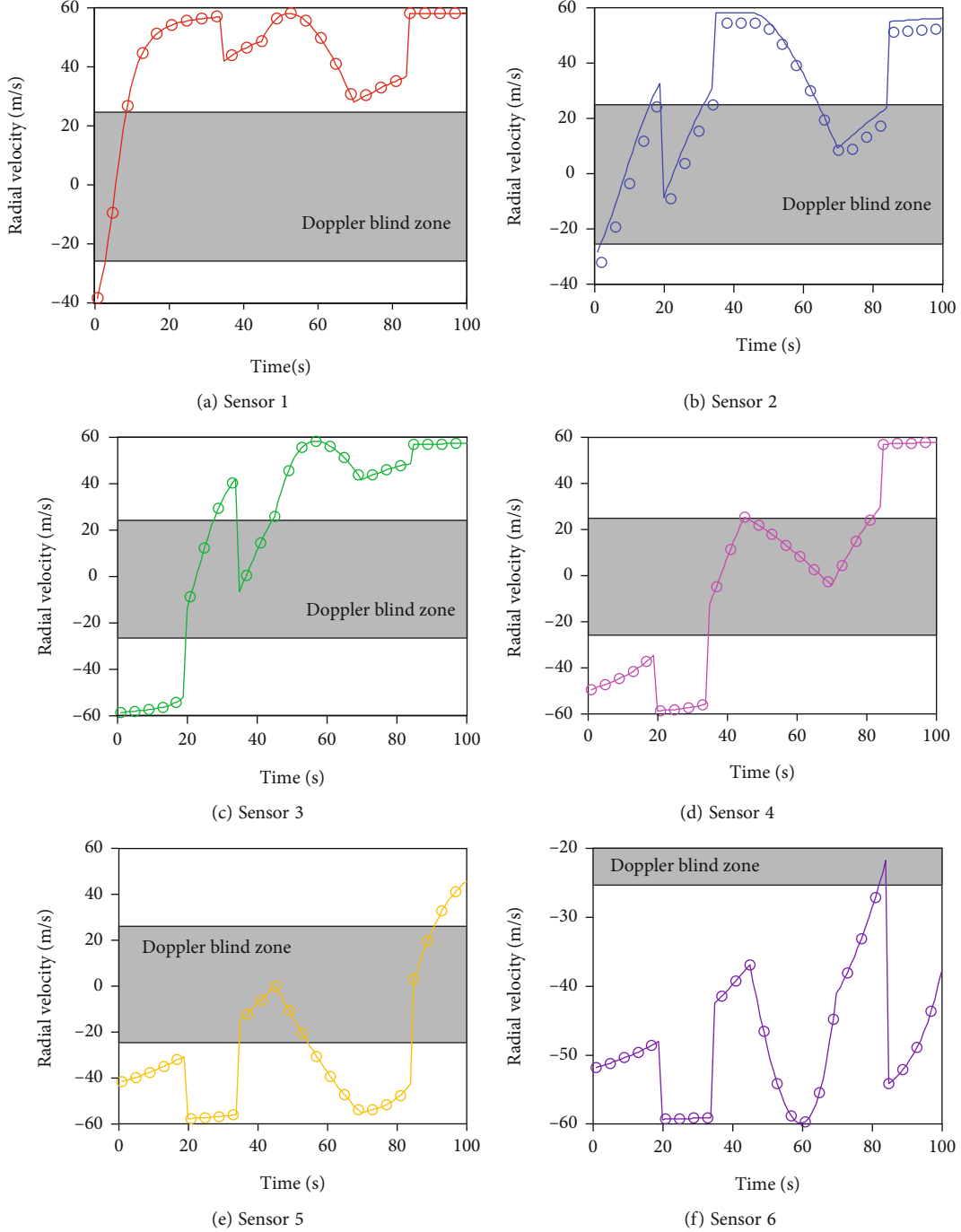
(e) Sensor 5

(f) Sensor 6

FIGURE 5: The radial velocities of the target relative to all sensors.

optimization function is established as follows.

$$\min \mathfrak{R}(A_{k:k+H-1}) = \underbrace{\mathrm{tr}\left[\mathbf{\Psi}(X \wedge_k, A_k)^{-1}\right]}_{\text{One-step revenue}} + \underbrace{\sum_{h=1}^{H-1} \gamma^h \mathrm{tr}\left[\mathbf{\Psi}(X \wedge_{k+h}, A_{k+h})^{-1}\right]}_{\text{Future expected revenue}},$$

$$\mathrm{s.t.} \begin{cases} \sum_{n=1}^{N} a_{k+h-1}^n = 1, 1 \le h \le H, \\ t(A_{k+h-1}) \ge t_{\min} \end{cases}$$

(47)

where $\mathrm{tr}\left[\mathbf{\Psi}(X \wedge_k, A_k)^{-1}\right]$ represents the trace of PCRLB after

executing sensor management scheme A_k and $\gamma(0 \le \gamma \le 1)$ is a discount coefficient that indicates. The optimal solution $A_{k:k+H-1}^{\mathrm{opt}}$ of the objective function is the best sensor management scheme in the time domain $[k, k + H - 1]$.

8. Numerical Simulations

As can be seen from Figure 4, in our simulations, six Doppler sensors are used to track a ground moving target that can move on the road or off-road. Besides, there are some obstacles in the battlefield which may obstruct the detection of the
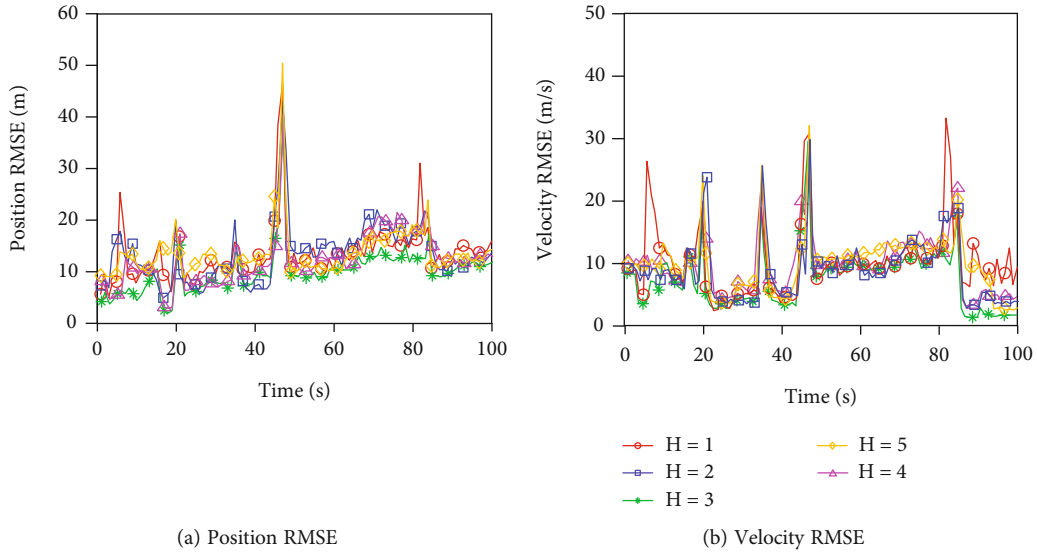
(a) Position RMSE

(b) Velocity RMSE

FIGURE 6: Comparisons of the RMSE under different decision steps.

(a) Position RTAMSE
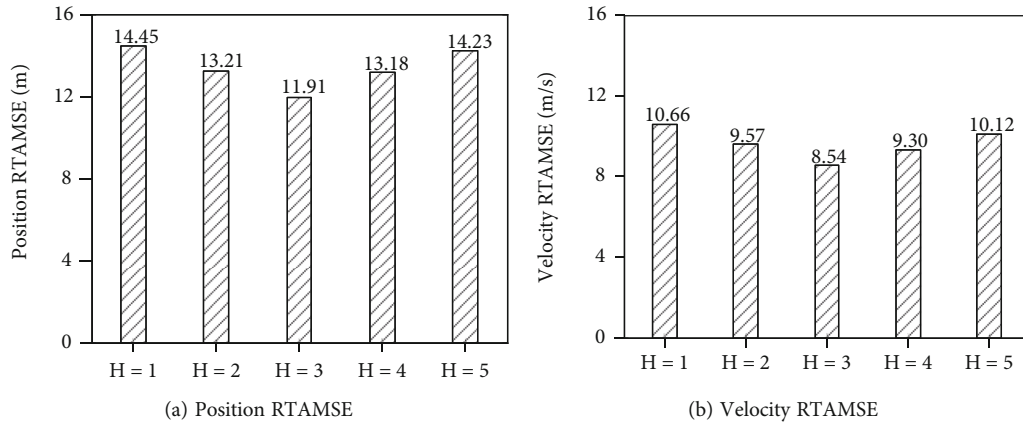
(b) Velocity RTAMSE

FIGURE 7: Comparison of the RTAMSE under different decision steps.

target. The sensor sampling interval is 1 s and the simulation duration is 100 s.

The initial position and velocity of the target are (0, 0) m and 60 m/s, respectively. The target moves off-road by turning right at an angle of 4° during 46-70 s, moves off-road in uniform straight line during 71-85 s, and maintains uniform straight line on road during other times. The process noise variance along the X and Y directions are 20 m and 20 m, and the process noise variances along the road and orthogonal to the road are 20 m and 1 m.

All sensors have the same MDV $V_{MDV} = 25$m/s, the minimum dwell time $t_{min} = 3$s, the false alarm probability $p_f = 10^{-6}$, the minimum detectable SNR $SNR_{min} = 15$db, and the maximum detection distance $R_{max} = 10$km. The other parameters of the sensors are shown in Table 1.

The other parameters used in simulations are displayed as follows: the discount coefficient $\gamma = 0.9$, the correlation coefficient ϖ of the measurement noise is 0.5, and the number of the unconstrained and DBZ particles is 100 in the PF-DBZ algorithm.

All the results were obtained by 100 trials of Monte Carlo simulations.

Figure 5 shows the radial velocities of the target relative to all sensors. Obviously, for each sensor, there is a situation where the radial velocity of the target is below than MDV, which means that the target will enter the DBZ of the sensor at some times and a single sensor cannot achieve continuous tracking of the target. Therefore, it is necessary for tracking the target to use an effective sensor management method.

8.1. Determination of the Decision Step. The decision step H is a very important parameter in sensor management, which can directly affect the effectiveness of target tracking [24]. Hence, the performances of target tracking under different H are studied in our simulations, and the corresponding result is used as a basis to determine the value of H. Figures 6 and 7 show the comparisons of the root mean square error (RMSE) and the root time average square error (RTAMSE) of the target under different H, respectively.

(a) PSMM

(b) MSMM

(c) CSMM

(d) FSMM

Doppler blind zone
Visual blind zone
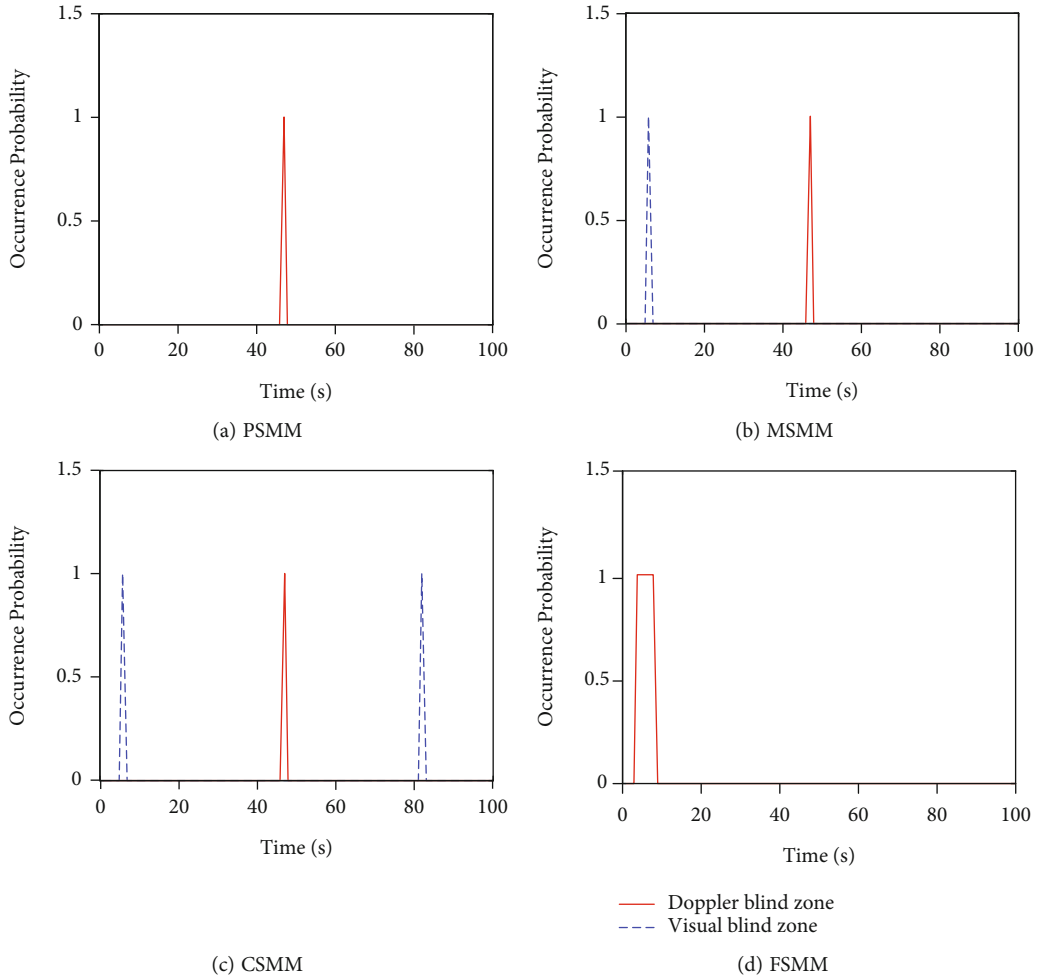
FIGURE 8: The occurrence probabilities of blind zones under different sensor management methods.

(a) Position RMSE

(b) Velocity RMSE

PSMM   MSMM
CSMM   FSMM
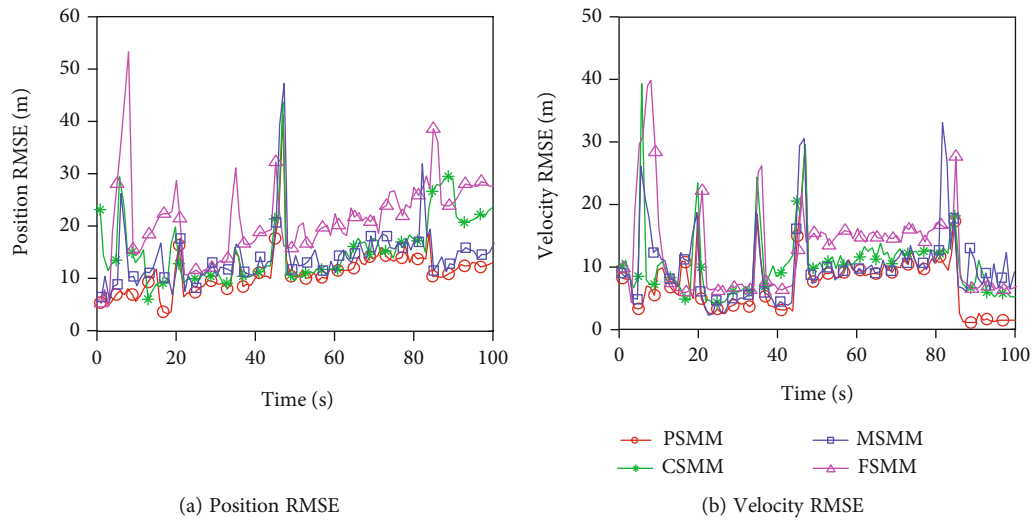
FIGURE 9: Comparisons of the RMSE under different sensor management methods.

As can be seen from Figures 6 and 7, when $H = 1 \sim 3$, with the increase of H, the RMSE and RTAMSE of the target decrease gradually, which means that the effect of tracking target is getting better and better. However, when $H > 3$, the corresponding tracking errors (RMSE and RTAMSE) increase instead of decreasing with the increase of H.

(a) Position RTAMSE

(b) Velocity RTAMSE
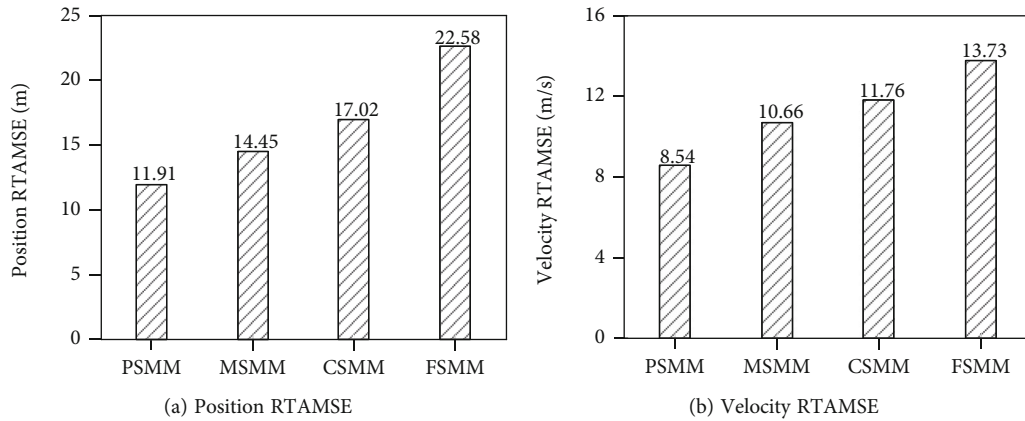
Figure 10: Comparisons of the RTAMSE of different sensor management methods.
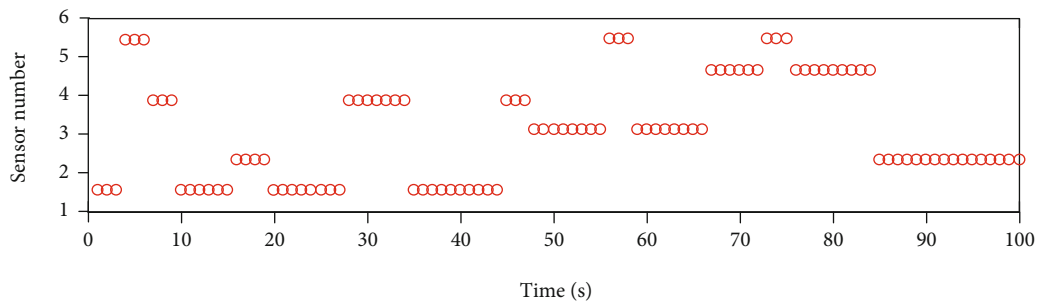
Figure 11: Sensor management schemes of the PSMM.

(a) Position RMSE

(b) Velocity RMSE
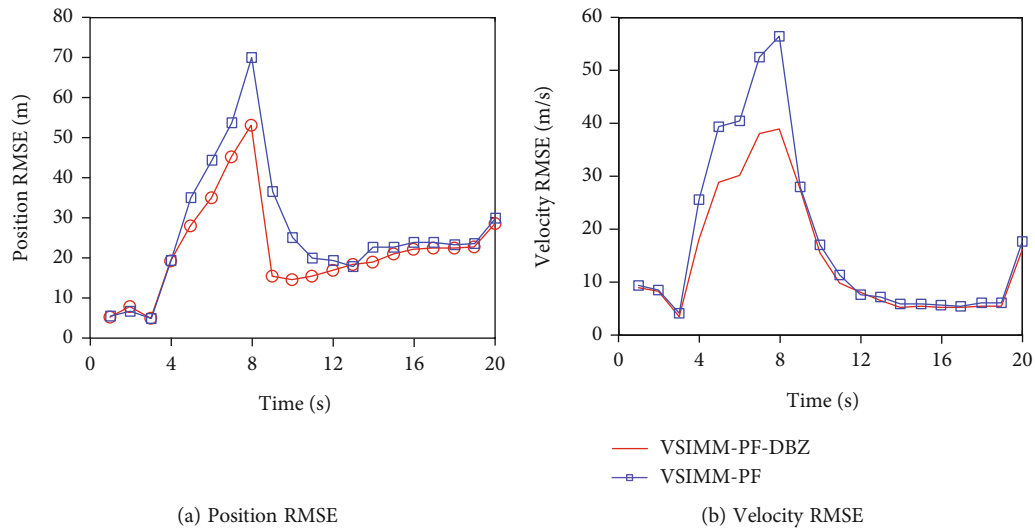
Figure 12: Comparisons of the RMSE of different target state estimation algorithms.

Especially when $H = 4$, the corresponding errors are larger than the errors under $H = 3$. The reason for these is that with the increase of H, the prediction error of the target state in the future will also increase, resulting in the increase of the target tracking error. At the same time, the occurrence of measurement uncertainty events such as the target enters the DBZ which will also lead to the increase of the tracking error in the multistep prediction. Therefore, it is not the case

that a larger decision step is better for tracking. Based on the above analysis, we choose $H = 3$ in next simulations.

8.2. Analysis of the Proposed Sensor Management Method. In order to clearly analyze the performance of the proposed sensor management method (PSMM) in this paper, we selected three existing sensor management methods to compare with it:

(a) Position RMSE

(b) Velocity RMSE
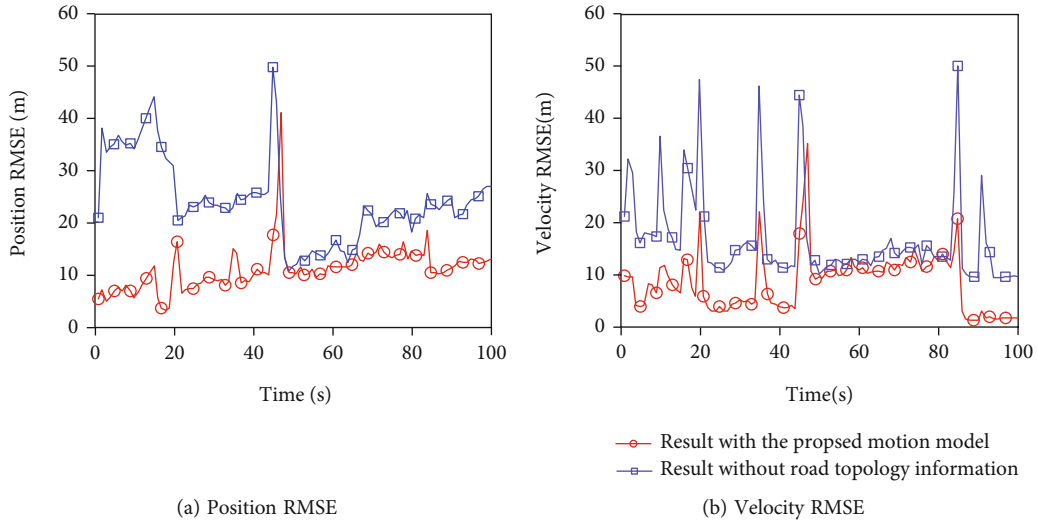
FIGURE 13: Comparisons of the RMSE under different motion models.

(a) Position RMSE

(b) Velocity RMSE
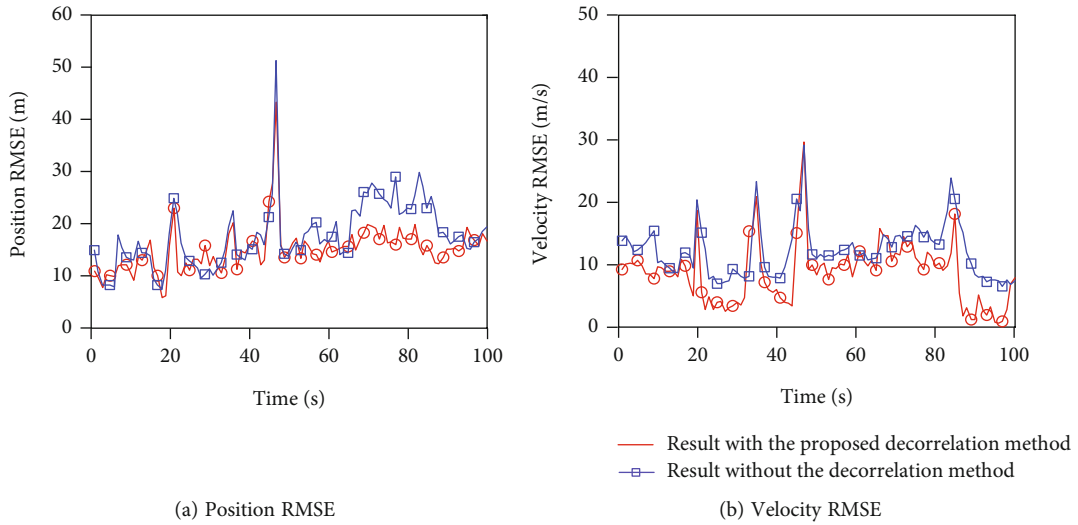
FIGURE 14: Comparisons of the RMSE under different measurement noise treatment methods ($\varpi = 0.1$).

(a) Position RMSE

(b) Velocity RMSE
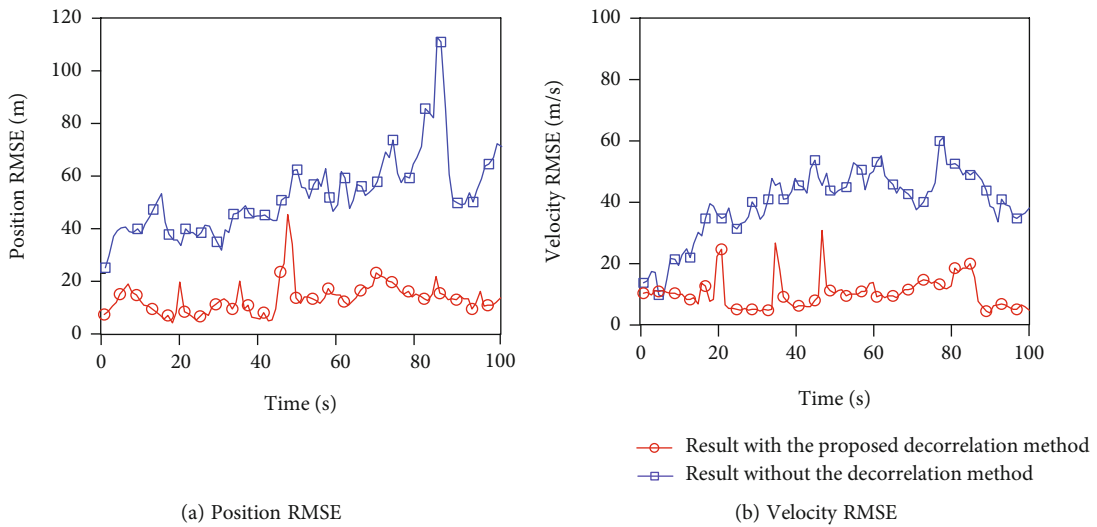
FIGURE 15: Comparisons of the RMSE under different measurement noise treatment methods ($\varpi = 0.8$).

(1) The myopic sensor management method (MSMM), in which the management scheme is decided based on one-step revenues in the future

(2) The closest sensor management method (CSMM), in which the sensor closest to the target is selected for tracking [19]

(3) The fixed sensor management method (FSMM), in which the fixed sensor is selected for tracking in the simulation duration (in our simulations, sensor 1 is selected)

The occurrence probabilities of blind zones under different sensor management methods in 100 repeated simulations are shown in Figure 8. For the convenience of presentation, we call the target to enter the visual blind zone (VBZ) when it is blocked by obstacles. Compared with the other methods, we can see that the DBZ and VBZ appear the least times under the PSMM. This is due to the PSMM considering not only the one-step revenue of the sensor system but also the future expected revenue in decision-making, which can better predict the target state and thus select the nonblind sensor for tracking. The MSM only considers the one-step revenue in decision-making which is myopic for tracking, so the corresponding blind zone occurrence probability is higher than those of PSMM. The CSMM and FSMM do not select the sensor based on the tracking revenue in the future, which makes the blind zones occurred many times in the tracking.

Figures 9 and 10 show the comparisons of the RMSE and RTAMSE under different sensor management methods, respectively. Obviously, the PSMM can obtain the best tracking performance compared with the other methods. Besides, the position RMSE curves and velocity RMSE curves under PSMM are relatively stable in the simulation duration, and the corresponding error is basically not sharply increased, which means that it can achieve continuous and stable tracking of the target by the PSMM.

Figure 11 shows the sensor management schemes of the PSMM, which is the optimal solution of the objective function (47). It can be seen that the minimum continuous working time of the selected sensor satisfies the constraint that the minimum dwell time $t_{\min} = 3 s$ in tracking, which verified that the management scheme is reasonable and effective.

Furthermore, in order to show that the proposed target state estimation algorithm based on the DBZ information is effective and advanced, the VSIMM-PF-DBZ algorithm is compared with the VSIMM-PF algorithm [43] in which the DBZ information is not be used. It can be seen from Figure 8(d) that the DBZ will appear continuously during 4-8 s under the FSMM. Therefore, we compare the two algorithms during 0-20 s under the FSMM in Figure 12, so as to study the tracking performance of them. Obviously, the tracking errors of the VSIMM-PF-DBZ are less than VSIMM-PF when the DBZ appears (4-8 s), which indicates that the estimation algorithm proposed in this paper can reasonably use the DBZ information to track the target.

8.3. Analysis of the Proposed Motion Model. In this paper, a more realistic target motion model is established based on road topology information. To analyze the effect of utilizing road topology information in terms of tracking performance, the proposed motion model is compared with the model without utilizing any road topology information. Note that if no information is used, equation (2) is always used to describe the target motion in proposed tracking algorithm. Figure 13 shows the comparisons of the RMSE under different motion models. Obviously, the position and velocity RMSE under the proposed motion model are less than those under the motion model without road topology information. Especially when the target moves on the road (0~45 s and 86~100 s), the difference between the results with the two kinds of models is obvious. From the comparisons, we can see that the tracking accuracy is improved by utilizing road topology information to describe the target motion.

8.4. Analysis of the Proposed Decorrelation Method of the Measurement Noise. To analyze the performance of the proposed decorrelation method of the measurement noise, the results of tracking errors corresponding to using and not using decorrelation methods are compared. Note that the correlation coefficient ϖ is set as two values, 0.1 and 0.8, and the other parameters remain unchanged.

Figures 14 and 15 show the comparison of the RMSE when $\varpi = 0.1$ and $\varpi = 0.8$, respectively. As can be seen, there is little difference between the results of the two methods when $\varpi = 0.1$. On the contrary, when $\varpi = 0.8$, the results with decorrelation method are much better than those without decorrelation method. The comparison results show that the proposed decorrelation method is effective, especially when the correlation coefficient is large.

9. Conclusions

In this paper, a sensor management method for ground moving target tracking is proposed, in which the DBZ information is used to improve the tracking accuracy in the presence of the DBZ. Firstly, a more realistic motion model of the ground target is established, in which the target motion state is divided into two categories: off-road and on-road. Secondly, a sensor measurement model under measurement uncertainty is presented, and a decorrelation method is proposed to solve the problem that the measurement noises are statistically correlated. Third, a target state estimation algorithm based on the DBZ information is proposed, in which the DBZ information is fully utilized. Furthermore, combined with the VSIMM, an estimation algorithm is given to track the maneuvering target with multiple models. Finally, the PCRLB is used to quantify the tracking accuracy in the future, and a nonmyopic sensor management optimization model is established. Simulation results indicate that the proposed sensor management method can track the target accurately by selecting the suitable sensor management scheme at each time. Meanwhile, the proposed motion model and decorrelation method of the measurement noise are verified to be effective. As future work, we will study the sensor management method for multitarget tracking in the presence of the DBZ.

Data Availability

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

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] M. Marcos, A. N. Vasconcelos, and M. Urbashi, "Optimal sensor management strategies in networked estimation," in *Proceedings of IEEE 56th Annual Conference Decision and Control*, pp. 5378–5385, Melbourne, VIC, Australia, December 2017.

[2] P. Salvagnini, F. Pernici, M. Cristani, G. Lisanti, A. Del Bimbo, and V. Murino, "Non-myopic information theoretic sensor management of a single pan-tilt-zoom camera for multiple object detection and tracking," *Computer Vision and Image Understanding*, vol. 134, pp. 74–88, 2015.

[3] H. G. Hoang and B. T. Vo, "Sensor management for multitarget tracking via multi-Bernoulli filtering," *Automatica*, vol. 50, no. 4, pp. 1135–1142, 2014.

[4] J. Z. HARE, S. GUPTA, and T. A. WETTERGREN, "POSE: prediction-based opportunistic sensing for energy efficiency in sensor networks using distributed supervisors," *IEEE Transactions on Cybernetics*, vol. 48, no. 7, pp. 2114–2127, 2018.

[5] J. Wu, Y. Zheng, Z. Sun, and H. Chen, "Multi-sensor management based on information increment and covariance of target priority discrimination," *Telecommunications and Radio Engineering*, vol. 79, no. 8, pp. 691–702, 2020.

[6] C. Pang and G. Shan, "Risk-based sensor scheduling for target detection," *Computers & Electrical Engineering*, vol. 77, no. 7, pp. 179–190, 2019.

[7] A. S. Chhetri, D. Morrell, and A. Papandreou-Suppappola, "Efficient search strategies for non-myopic sensor scheduling in target tracking," in *Conference Record of the Thirty-Eighth Asilomar Conference on Signals, Systems and Computers, 2004*, pp. 2106–2110, Pacific Grove, CA, USA, November 2004.

[8] N. Cao, S. Choi, E. Masazade, and P. K. Varshney, "Sensor selection for target tracking in wireless sensor networks with uncertainty," *IEEE Transactions on Signal Processing*, vol. 64, no. 20, pp. 5191–5204, 2016.

[9] A. S. Chhetri, D. Morrell, and A. Papandreou-Suppappola, "On the use of binary programming for sensor scheduling," *IEEE Transactions on Signal Processing*, vol. 55, no. 6, pp. 2826–2839, 2007.

[10] X. S. Liu, L. Zhou, and X. Y. Du, "A method of sensor management based on target priority and information gain," *Acta Electronica Sinica*, vol. 33, no. 9, pp. 1683–1687, 2005.

[11] K. White, "Integrated tracking and sensor management based on expected information gain," in *In Proceedings of SPIE-The International Society for Optical Engineering*, Orlando, FL, USA, April 2007.

[12] X. Miao, J. Hu, G. Song, and A. Y. Zomaya, "Distributed segment-based anomaly detection with Kullback-Leibler divergence in wireless sensor networks," *IEEE Transactions on Information Forensics and Security*, vol. 12, no. 1, pp. 101–110, 2017.

[13] Q. Liu and Z. Liu, "A method of maneuvering target collaboration tracking based on Renyi information gain," *Control and Decision*, vol. 27, no. 9, pp. 1437–1440, 2012.

[14] C. M. Kreucher, A. O. Hero, and K. Kastella, "A comparison of task driven and information driven sensor management for target tracking," in *Proceedings of the 44th IEEE Conference on Decision and Control*, pp. 4004–4009, Seville, Spain, December 2005.

[15] F. Katsilieris, H. Driessen, and A. Yarovoy, "Threat-based sensor management for target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 51, no. 4, pp. 2772–2785, 2015.

[16] C. Pang, G. Shan, X. Duan, and G. Xu, "A multi-mode sensor management approach in the missions of target detecting and tracking," *Electronics*, vol. 8, no. 1, p. 71, 2019.

[17] S. Martin, "Risk-based sensor resource management for field of view constrained sensors," in *2015 18th International Conference on Information Fusion (Fusion)*, 2048, p. 2041, Washington, DC, USA, July 2015.

[18] C. Shi, W. Fei, S. Mathini, and J. Zhou, "LPI optimization framework for target tracking in radar network architectures using information-theoretic criteria," *International Journal of Antennas & Propagation*, vol. 2014, pp. 1–10, 2014.

[19] Z. N. Zhang and G. L. Shan, "UTS-based foresight optimization of sensor scheduling for low interception risk tracking," *International Journal of Adaptive Control & Signal Processing*, vol. 28, no. 10, pp. 921–931, 2014.

[20] V. Krishnamurthy, "Emission management for low probability intercept sensors in network centric warfare," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 133–152, 2005.

[21] M. Kalandros and L. Pao, "Multisensor covariance control strategies for reducing bias effects in interacting target scenarios," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 41, no. 1, pp. 153–173, 2005.

[22] C. Yang, L. Kaplan, and E. Blasch, "Performance measures of covariance and information matrices in resource management for target state estimation," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 3, pp. 2594–2613, 2012.

[23] J. M. Dias and P. A. Marques, "Multiple moving target detection and trajectory estimation using a single SAR sensor," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 2, pp. 604–624, 2003.

[24] G. Xu, G. Shan, and X. Duan, "Non-myopic scheduling method of mobile sensors for manoeuvring target tracking," *IET Radar, Sonar & Navigation*, vol. 13, no. 11, pp. 1899–1908, 2019.

[25] A. Mohammadi and A. Asif, "Consensus-based distributed dynamic sensor selection in decentralised sensor networks using the posterior Cramer-Rao lower bound," *Signal Processing*, vol. 108, pp. 558–575, 2015.

[26] A. K. Gostar, R. Hoseinnezhad, T. Rathnayake, X. Wang, and A. Bab-Hadiashar, "Constrained sensor control for labeled multi-Bernoulli filter using Cauchy-Schwarz divergence," *IEEE Signal Processing Letters*, vol. 24, no. 9, pp. 1313–1317, 2017.

[27] K. Wan, X. Gao, L. I. Bo, and L. I. Fei, "Using approximate dynamic programming for multi-ESM scheduling to track ground moving targets," *Journal of Systems Engineering & Electronics*, vol. 29, no. 1, pp. 74–85, 2018.

[28] Y. Cheng and T. Singh, "Efficient particle filtering for road-constrained target tracking," *IEEE Transactions on Aerospace & Electronic Systems*, vol. 43, no. 99, pp. 1454–1469, 2007.

[29] Y. Miao, C. Liu, W. H. Chen, and J. Chambers, "A Bayesian framework with an auxiliary particle filter for GMTI-based ground vehicle tracking aided by domain knowledge," in *In Proceedings of SPIE-The International Society for Optical Engineering*, Baltimore, Maryland, United States, June 2014.

[30] M. Ulmke and W. Koch, "Road-map assisted ground moving target tracking," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 42, no. 4, pp. 1264–1274, 2006.

[31] W. Koch, J. Koller, and M. Ulmke, "Ground target tracking and road map extraction," *Isprs Journal of Photogrammetry & Remote Sensing*, vol. 61, no. 3-4, pp. 197–208, 2006.

[32] Hyondong Oh, Seungkeun Kim, and A. Tsourdos, "Road-map-assisted standoff tracking of moving ground vehicle using nonlinear model predictive control," *IEEE Transactions on Aerospace & Electronic Systems*, vol. 51, no. 2, pp. 975–986, 2015.

[33] Z. Xinyan and Z. Wei, "Application of road information in ground moving target tracking," *Chinese Journal of Aeronautics*, vol. 20, no. 6, pp. 529–538, 2007.

[34] J. Zheng and M. Gao, "Tracking ground targets with a road constraint using a GMPHD filter," *Sensors*, vol. 18, no. 8, pp. 2723–2743, 2018.

[35] M. Arulampalam, N. Gordon, M. Orton, and B. Ristic, "A variable structure multiple model particle filter for GMTI tracking," in *Proceedings of the 15th International Conference on Information Fusion*, pp. 927–934, Annapolis, MD, USA, July 2002.

[36] X. Gongguo, S. Ganlin, and D. Xiusheng, "Sensor scheduling for ground maneuvering target tracking in presence of detection blind zone," *Journal of Systems Engineering and Electronics*, vol. 31, no. 4, pp. 692–702, 2020.

[37] W. Wu, W. Liu, J. Jiang, L. Gao, Q. Wei, and C. Liu, "GM-PHD filter-based multi-target tracking in the presence of Doppler blind zone," *Digital Signal Processing*, vol. 52, pp. 1–12, 2016.

[38] W. Wu, H. Sun, Y. Cai, and J. Xiong, "MM-GLMB filter-based sensor control for tracking multiple maneuvering targets hidden in the Doppler blind zone," *IEEE Transactions on Signal Processing*, vol. 68, pp. 4555–4567, 2020.

[39] H. Huang, R. Yang, P. Foo et al., "Convoy tracking in Doppler blind zone regions using GMTI radar," in *Proceedings of the 16th International Conference on Information Fusion*, pp. 1768–1775, Istanbul, Turkey, July 2013.

[40] M. Mertens and U. Nickel, "GMTI tracking in the presence of Doppler and range ambiguities," in *Proceedings of the 14th International Conference on Information Fusion*, pp. 1369–1376, Chicago, Illinois, USA, July 2011.

[41] M. Lei and C. Han, "Sequential nonlinear tracking using UKF and raw range-rate measurements," *IEEE Transactions on Aerospace & Electronic Systems*, vol. 43, no. 1, pp. 239–250, 2007.

[42] M. Byrne, K. White, and J. Williams, "Rolling horizon non-myopic scheduling of multifunction radar for search and track," in *International Conference on Information Fusion*, pp. 634–642, Heidelberg, Germany, July 2016.

[43] M. Yu, H. Oh, and W. Chen, "An improved multiple model particle filtering approach for manoeuvring target tracking using airborne GMTI with geographic information," *Aerospace Science and Technology*, vol. 52, no. 5, pp. 62–69, 2016.

[44] B. R. Mahafza, *Radar Systems Analysis and Design Using MATLAB*, Publishing House Electronic Industry, Beijing, China, 2008.

[45] C. Gao and W. Chen, "Ground moving target tracking with VS-IMM using mean shift unscented particle filter," *Chinese Journal of Aeronautics*, vol. 24, no. 5, pp. 622–630, 2011.

Research Article

A GMM-Based Secure State Estimation Approach against Dynamic Malicious Adversaries

Cui Zhu (ID), Zile Wang (ID), Zeyuan Zang, Yuxuan Li, and Huanming Zheng

Beijing Information Science and Technology University, School of Information and Communication Engineering, Beijing 100192, China

Correspondence should be addressed to Cui Zhu; cuizhu_lzy@bistu.edu.cn

We consider the secure state estimation of linear time-invariant Gaussian systems subject to dynamic malicious attacks. An error compensator is proposed to reduce the impact of local error data on state estimation. Based on that, a new estimation algorithm based on the Gaussian mixture model (GMM) aiming at dynamic attacks is proposed, which can cluster the local state estimates autonomously and improve the remote estimation accuracy effectively. The superiority of the proposed algorithm is verified by numerical simulations.

1. Introduction

Cyberphysical systems (CPSs), such as transportation networks and smart grids, integrate sensing, computing, and control technologies with a communication infrastructure. Tight integration and cooperation between cyber and physical components are the features of CPSs [1]. However, CPSs are vulnerable to any successful attacks especially network attacks on the data and communication channels, which causes serious harms to the national economy and social security, for example, the Stuxnet storm reported in [2], StuxNet malware [3], power blackouts in Brazil [4], and Maroochy water bleach [5]. Due to the widespread application of CPSs in many real-life critical infrastructures [6], the security of CPSs has become an increasingly important issue which has attracted attention from many researchers in the past decades.

In the recent literature, the secure state estimation is an important research direction of CPSs security. In [7], a distributed state estimation method based on parallelized stream computing is proposed, which can not only significantly improve the speed of state estimation calculation but also reduce the interregional convergence correlation and the residual pollution. In [8], a new sequential estimation method is proposed to improve the estimation accuracy,

which sequentially estimates states by the particle filter (PF) and parameters by the separable natural evolution strategy (SNES). The state estimation of three-phase power system models is studied in [9]. In [10], a Bayesian network based on the wireless power transfer (WPT) system state estimation algorithm is proposed, which can estimate the WPT system states in a distributed way using the Bayesian tree structure. In [11], a robust generalized maximum likelihood (GM) estimator, which leverages modified projection statistics and a Huber convex score function, is designed to bound the influence of observation outliers while maintaining its high statistical estimation efficiency. In [12], a distributed dynamic state estimation method for microgrids incorporating distributed energy resources is presented. In [13], a robust generalized maximum-likelihood Koopman operator-based Kalman filter (GM-KKF) is designed, which can estimate the rotor angle and speed of synchronous generators. In [14], a correlation-aided robust adaptive unscented Kalman filter (UKF) for power system decentralized dynamic state estimation with unknown inputs is presented, which has lower requirement of number of measurements for dynamic state estimation while achieving better robustness against bad data. In [15, 16], the state estimation method based on undamaged sensors is studied. In [17, 18], the state estimation for different systems is studied based on the

convex optimization methods. In [19], by modeling and adopting a variety of models, a random Bayesian approach is proposed to solve the state estimation against switching patterns and signal attacks. In [20], the state estimation against fixed target attacks, switched target attacks with disturbance, and sparse sensor attacks are considered, and the sufficient condition for the existence of the switched observer is given. In [21], a fusion algorithm based on the Gaussian mixture model is presented to solve the estimation of a linear time-invariant Gaussian system under stealth attacks. However, the dynamic attacks are not considered. In [22], a dynamic combination strategy and a distributed Kalman filter are proposed, which improve the robustness of the system against random error data injection and replay attacks.

Most of the studies mentioned above have focused on static attacks. However, dynamic attacks are very common in real systems. Therefore, this paper considers the state estimation for a networked system suffering from dynamic adversaries as shown in Figure 1. The different sensors are attacked randomly at each time instant, and it is assumed that the number of attached sensors does not exceed half of the sensors.

Inspired by [21], we have designed an error compensator to reduce the impact of incorrect data on state estimation. Based on that, a new GMM-based state estimation algorithm is presented, which can effectively improve the state estimation accuracy against the dynamic adversaries. The contributions of this article are listed as follows:

(1) A new error compensator is proposed to alleviate the influence of wrong data on state estimation, which can judge whether the beliefs generated by the expectation-maximum (EM) algorithm are accurate based on the observability of the system, and correct the doubtful beliefs

(2) By introducing the error compensator, a new GMM-based estimation algorithm is presented, which can improve the estimation accuracy effectively. The proposed algorithm can fuse the local data by adopting the modified beliefs as the weights of the local data with the centralized Kalman filter

The rest of the paper is organized as follows. Section II formulates the model of the considered system and the problem of interest. Section III proposes the error compensator and the new GMM-based state estimation algorithm against dynamic adversaries. In Section IV, the effectiveness of the proposed algorithm is demonstrated by numerical simulations. Conclusions are given in Section V.

Notation: $\mathbb{N}$ and $\mathbb{R}$ are the sets of positive integers and real numbers, respectively. $\mathbb{R}^n$ denotes the n-dimensional Euclidean space. $\mathbb{S}_+^n(\mathbb{S}_{++}^n)$ is the set of $n \times n$ positive semidefinite (definite) matrices. We write $X \geq 0 (X > 0)$ when $X \in \mathbb{S}_+^n(\mathbb{S}_{++}^n)$. X' denotes the transpose of matrix X. $\mathbb{E}[\cdot]$ is the expectation of a random variable. $\mathcal{N}(\mu, \Sigma)$ is the Gaussian distribution with mean μ and covariance matrix Σ, and $X \sim \mathcal{N}(\mu, \Sigma)$ denotes X follows the Gaussian distribution $\mathcal{N}(\mu, \Sigma)$. $Diag\{\cdot\}$ denotes a block diagonal matrix.
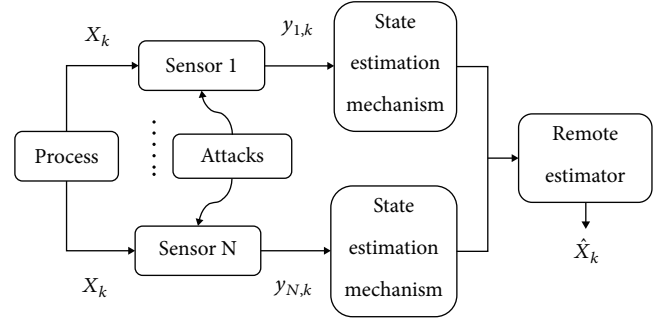
Figure 1: The networked system under attacks.

2. Problem Formulation

Consider the following networked system under attacks:

$$x_{k+1} = Ax_k + w_k, \tag{1}$$

$$y_{i,k} = C_i x_k + v_{i,k} + a_{i,k}, \tag{2}$$

where $x_k \in \mathbb{R}^n$ denotes the system state, $y_{i,k} \in \mathbb{R}^{m_i}$ represents the measured value from sensor i at time k, and $a_{i,k} \in \mathbb{R}^{m_i}$ is attack signal. The number of sensors is denoted by N. $w_k \in \mathbb{R}^n$ is the process noise, and $w_k \sim \mathcal{N}(0, Q)$. $v_{i,k} \in \mathbb{R}^{m_i}$ is the measurement noise, and $v_{i,k} \sim \mathcal{N}(0, R_i)$. Meanwhile, it is assumed that $\mathbb{E}[w_k w_l'] = \delta_{kl} Q (Q \geq 0)$, $\mathbb{E}[v_{i,k} v_{j,l}'] = \delta_{ij} \delta_{kl} R_i (R_i > 0)$, where $i = j(i \neq j)$, $\delta_{i,j} = 1(\delta_{i,j} = 0)$. $\mathbb{E}[w_k v_{i,l}'] = 0$, $\forall k, l \in \mathbb{N}$, $i, j = 1, 2, \cdots, N$. The initial state x_0 is independent of w_k and $v_{i,k}$ for all $k \geq 0$ and $x_0 \sim \mathcal{N}(0, \Pi_0)$. (A, C_i) and $(A, \sqrt{Q})$ are detectable and controllable, respectively.

The malicious attack $a_{i,k} \in \mathbb{R}^{m_i}$ satisfies the following assumptions:

Assumption 1. Any $s (s \leq N/2)$ sensors can be corrupted by the adversary, and the output values of the sensors are changed. Only when sensor i is unattacked, $a_{i,k} = 0$.

Assumption 2. The number of attacked sensors s is unknown, stochastic, and variable.

Assumption 3. The system parameters and noise statistics are known for the adversary.

Assumption 4. $a_{i,k}$ is statistically independent of $\{w_{\mathcal{K}}\}_{\mathcal{K}>k}$ and $\{v_{i,\mathcal{K}}\}_{\mathcal{K}>k}$, respectively.

Remark 1. According to [23, 24], it is impossible to accurately reconstruct the state of a system when more than half the sensors are attacked. Thus, we assume that the maximum number of damaged sensors does not exceed N/2 in this paper, i.e., the upper limit of s is N/2.

When the system is not attacked, the measurements at time instant k can be stacked as

$$y_k = Cx_k + v_k, \tag{3}$$

where

$$
\begin{aligned}
y_k &\triangleq \left[y'_{1,k} y'_{2,k} \cdots y'_{N,k} \right]', \\
v_k &\triangleq \left[v'_{1,k} v'_{2,k} \cdots v'_{N,k} \right]', \\
C &\triangleq \left[C'_1 C'_2 \cdots C'_N \right]', \\
R &\triangleq \mathrm{Diag}\{ R_1, R_2, \cdots, R_N \}.
\end{aligned}
\tag{4}
$$

Then, we adopt a centralized Kalman filter as the remote estimator:

$$
\begin{aligned}
\hat{x}_k^- &= A\hat{x}_{k-1}, \\
P_k^- &= A P_{k-1} A' + Q, \\
K_k &= P_k^- C' \left(C P_k^- C' + R \right)^{-1}, \\
\hat{x}_k &= \hat{x}_k^- + K_k (y_k - C\hat{x}_k^-), \\
P_k &= (I - K_k C) P_k^-,
\end{aligned}
\tag{5}
$$

where $\hat{x}_k^-$ and $\hat{x}_k$ are the priori and the posteriori estimation of the system state x_k, respectively. P_k^- and P_k are the priori and posteriori estimation error covariance, respectively. K_k is the Kalman filter gain.

From [21], we know that the information-form Kalman filter can be expressed as

$$\hat{x}_k = \hat{x}_k^- + P_k C' R^{-1}(y_k - C\hat{x}_k^-), \tag{6}$$

$$(P_k)^{-1} = (P_{k-1}^-)^{-1} + C' R^{-1} C. \tag{7}$$

Similarly, the local Kalman filter for sensor i can be written as

$$
\begin{aligned}
\hat{x}_{i,k} &= \hat{x}_{i,k}^- + P_{i,k} C'_i R_i^{-1} \left(y_{i,k} - C_i \hat{x}_{i,k}^- \right), \\
(P_{i,k})^{-1} &= \left(P_{i,k-1}^- \right)^{-1} + C'_i R_i^{-1} C_i.
\end{aligned}
\tag{8}
$$

It is noted that P_k and $P_{i,k}$ can be calculated offline. According to [25], the Kalman filter converges from any initial condition exponentially when (A, C_i) and $(A, \sqrt{Q})$ are detectable and controllable, respectively. The steady-state values of local and centralized Kalman filter are defined as

$$
\begin{aligned}
\bar{P}_i &\triangleq \lim_{k \to +\infty} P_{i,k}, \quad \bar{P}_i^- \triangleq \lim_{k \to +\infty} P_{i,k}^-, \\
P &\triangleq \lim_{k \to +\infty} P_k, \quad P^- \triangleq \lim_{k \to +\infty} P_k^-.
\end{aligned}
\tag{9}
$$

It is assumed that the system starts from the steady state with $P_{i,0} = \bar{P}_i$ and $P_0 = P$, and the fixed-gain of local and centralized Kalman filters can be represented as:

$$
\begin{aligned}
K_i &= \bar{P}_i C'_i R_i^{-1} = \bar{P}_i^- C'_i \left(C_i \bar{P}_i^- C'_i + R_i \right)^{-1}, \\
K &= P C' R^{-1} = P^- C' \left(C P^- C' + R \right)^{-1}.
\end{aligned}
\tag{10}
$$

The objective of this paper is to design a new GMM-based estimation method for systems suffering from dynamic adversaries.

3. The GMM-Based State Estimation

In this section, an error compensator and the GMM-based state estimation algorithm against dynamic adversaries are proposed.

3.1. Modeling and the EM Algorithm. For a Gaussian mixture model with $\mathbb{Q}$ components [21], the mean and covariance of the q-th component $\mathbb{Q}_q$ $(q \in \{1, 2, \cdots, \mathbb{Q}\})$ are expressed as $\mu^{(q)}$ and $\Sigma^{(q)}$, respectively. $\pi^{(q)}$ is the mixture component weights of $\mathbb{Q}_q$, and $\sum_{q=1}^{\mathbb{Q}} \pi^{(q)} = 1$. In this case, the mixture density of a Gaussian mixture model can be expressed as

$$p(x) = \sum_{q=1}^{\mathbb{Q}} p(x \mid \mathbb{Q}_q) \, \mathrm{Pr}(\mathbb{Q}_q) = \sum_{q=1}^{\mathbb{Q}} \pi^{(q)} f\left(x ; \mu^{(q)}, \Sigma^{(q)}\right), \tag{11}$$

where $p(x \mid \mathbb{Q}_q)$ and $\mathrm{Pr}(\mathbb{Q}_q)$ are the Gaussian distribution density and weight of the q-th component, respectively. Function $f(x ; \mu, \Sigma)$ is the probability density function (pdf) for Gaussian random variables:

$$f(x ; \mu, \Sigma) = \frac{1}{\sqrt{(2\pi)^n |\Sigma|}} \exp\left(-\frac{1}{2} (x - \mu)' \Sigma^{-1} (x - \mu) \right). \tag{12}$$

At time instant k, we denote the means of the state variables for sensor i as $\mu_k^{(1)}$ under the unattacked scenario and $\mu_k^{(2)}$ under the attacked-scenario, respectively. $\Sigma_k^{(1)}$ and $\Sigma_k^{(2)}$ represent the covariance when sensor i is unattacked and attacked, respectively. The local state estimation $\hat{x}_{i,k}$ follows different distributions depending on whether sensor i is attacked or not. According to the definition of GMM and the analysis of Kalman filtering in [25], it can be known that when sensor i is unattacked (defined as the first component),

$\widehat{x}_{i,k}$ follows the Gaussian distribution with the mean $\mu_k^{(1)}$ and the fixed covariance $\Sigma_k^{(1)} = \bar{P}_i$, i.e., $p(\widehat{x}_{i,k} \mid \mathcal{Q}_1) \sim \mathcal{N}(\mu_k^{(1)}, \bar{P}_i)$, $\forall i \in N$. When sensor i is attacked (defined as the second component), the exact distribution of $\widehat{x}_{i,k}$ is unknown since the specific type and the starting time of attacks are unknown. In this case, similar to [21], we can adopt a Gaussian distribution with the first and second moments, i.e., $p(\widehat{x}_{i,k} \mid \mathcal{Q}_2) \sim \mathcal{N}(\mu_k^{(2)}, \Sigma_k^{(2)})$, $\forall i \in N$, to approximate the distribution of all local estimates in the second component. Then, $\widehat{x}_{i,k}$ can be described by the following 2-component Gaussian mixture model:

$$
\begin{aligned}
p(\widehat{x}_{i,k}) &= \sum_{q=1}^{2} p(\widehat{x}_{i,k} \mid \mathcal{Q}_q) \Pr (\mathcal{Q}_q) \\
&= \pi_k^{(1)} p(\widehat{x}_{i,k} \mid \mathcal{Q}_1) + \pi_k^{(2)} p(\widehat{x}_{i,k} \mid \mathcal{Q}_2) \\
&= \pi_k^{(1)} f\left(\widehat{x}_{i,k}; \mu_k^{(1)}, \bar{P}_i\right) + \pi_k^{(2)} f\left(\widehat{x}_{i,k}; \mu_k^{(2)}, \Sigma_k^{(2)}\right),
\end{aligned}
\tag{13}
$$

where $\pi_k^{(1)}$ and $\pi_k^{(2)}$ are the weights of the first and second components at time k, respectively.

The observation data set is defined as $\mathcal{Z}_k = \{\widehat{x}_{i,k}\}_{i=1}^{N}$. According to [26, 27], it is known that the expectation-maximization (EM) algorithm can be adopted to find the maximum likelihood estimates for the parameter $\Phi_k = \{\pi_k^{(q)}, \mu_k^{(q)}, \Sigma_k^{(2)}\}_{q=1}^{2}$ using $\mathcal{Z}_k = \{\widehat{x}_{i,k}\}_{i=1}^{N}$. The log likelihood is shown as

$$
\begin{aligned}
\mathcal{L}(\Phi_k; \mathcal{Z}_k) = \sum_{i=1}^{N} \log \Big(&\pi_k^{(1)} f\left(\widehat{x}_{i,k}; \mu_k^{(1)}, \bar{P}_i\right) \\
&+ \pi_k^{(2)} f\left(\widehat{x}_{i,k}; \mu_k^{(2)}, \Sigma_k^{(2)}\right) \Big).
\end{aligned}
\tag{14}
$$

Generally, the EM algorithm is divided into two steps: the expectation and maximization step. First, initializing the parameter Φ_k at each time k, then the expectation step generates a belief $\gamma_{i,k}^{(q)}(q = 1, 2)$ based on Φ_k and $\widehat{x}_{i,k}$ for each sensor:

$$
\gamma_{i,k}^{(1)} = \frac{\pi_k^{(1)} f\left(\widehat{x}_{i,k}; \mu_k^{(1)}, \Sigma_k^{(1)}\right)}{\pi_k^{(1)} f\left(\widehat{x}_{i,k}; \mu_k^{(1)}, \Sigma_k^{(1)}\right) + \pi_k^{(2)} f\left(\widehat{x}_{i,k}; \mu_k^{(2)}, \Sigma_k^{(2)}\right)},
\tag{15}
$$

$$
\gamma_{i,k}^{(2)} = 1 - \gamma_{i,k}^{(1)},
\tag{16}
$$

where $\gamma_{i,k}^{(1)}$ and $\gamma_{i,k}^{(2)}$ represent the probability of sensor i belonging to the component $\mathcal{Q}_1$ and $\mathcal{Q}_2$, respectively.

Given all beliefs $\gamma_{i,k}^{(1)}$ and $\gamma_{i,k}^{(2)}$, the parameters $\{\pi_k^{(q)}, \mu_k^{(q)}, \Sigma_k^{(2)}\}_{q=1}^{2}$ are reestimated in the maximization step:

$$
\pi_k^{(q)} = \frac{\sum_{i=1}^{N} \gamma_{i,k}^{(q)}}{N},
\tag{17}
$$

$$
\mu_k^{(q)} = \frac{\sum_{i=1}^{N} \gamma_{i,k}^{(q)} \widehat{x}_{i,k}}{\sum_{i=1}^{N} \gamma_{i,k}^{(q)}},
\tag{18}
$$

$$
\Sigma_k^{(2)} = \frac{\sum_{i=1}^{N} \gamma_{i,k}^{(2)} \left(\widehat{x}_{i,k} - \mu_k^{(2)}\right) \left(x \wedge_{i,k} - \mu_k^{(2)}\right)'}{\sum_{i=1}^{N} \gamma_{i,k}^{(2)}}.
\tag{19}
$$

The expectation and maximization steps iterate until they converge to a certain value. This iterative procedure maximizes the concave lower bound of the log likelihood in (14).

3.2. The Error Compensator. In this subsection, an error compensator is proposed to reduce the influence of incorrect data on the state estimation.

According to 3.1, the EM algorithm can be used to calculate the GMM parameters and find the maximum likelihood estimation. However, the convergence and clustering results of the EM algorithm are affected by the initial parameters. In this paper, the first and second moments are adopted as the initial parameters of the second cluster. Due to the randomness of dynamic adversary and its specific type is unknown, the output of some attacked sensors may be similar to that of normal sensors at some moments. In this case, $\gamma_{i,k}^{(1)}$ will be miscalculated as $\gamma_{i,k}^{(2)}$ in the iterative process (15)-(19), since the observed data are considered to be closer to the second cluster by the EM algorithm. When the above case occurs, the estimation accuracy will be reduced seriously because the number of data available for fusion is less than $N/2$. On the other hand, the measurements that are similar to the true measurements can provide useful information for the remote state estimation, which means that the data belonging to the second cluster can be adopted to estimate system state. Hence, a compensator is designed to solve the above problem.

$\bar{\gamma}_k^{(2)}$ represents the average of all $\gamma_{i,k}^{(2)}$ at time instant k, which can be calculated as follows:

$$
\bar{\gamma}_k^{(2)} = \frac{\sum_{i=1}^{N} \gamma_{i,k}^{(2)}}{N}.
\tag{20}
$$

According to the EM algorithm, $\gamma_{i,k}^{(2)}$ tends to 1 if and only if sensor i is attacked, and the expectation step is accurate, which causes $\sum_{i=1}^{N} \gamma_{i,k}^{(2)}$ to approach s. When the expectation step is miscalculated, $\sum_{i=1}^{N} \gamma_{i,k}^{(2)}$ tends to $N - s$ since $\gamma_{i,k}^{(2)}$ approachs 0 for the attacked sensor i. According to Assumptions 1–4, the maximum number of damaged sensors does not exceed $N/2$ (namely, $s \leq N/2$), which means $N - s > N/2$. Hence, it can be known that $\gamma_{i,k}^{(1)}$ and $\gamma_{i,k}^{(2)}$ are miscalculated if $\sum_{i=1}^{N} \gamma_{i,k}^{(2)} > N/2$. Based on the above analysis, the compensator is designed as follows:

$$
\widehat{\gamma}_{i,k}^{(1)} = \begin{cases} \gamma_{i,k}^{(1)}, & \bar{\gamma}_k^{(2)} \leq \varepsilon \\ \gamma_{i,k}^{(2)}, & \bar{\gamma}_k^{(2)} > \varepsilon \end{cases},
$$

$$
\widehat{\gamma}_{i,k}^{(2)} = 1 - \widehat{\gamma}_{i,k}^{(1)},
\tag{21}
$$

```
1 // Run Kalman filter to steady state.
2: Initialize $\hat{x}_{i,-\infty} = 0, P_{i,-\infty} = \Pi_i, \hat{x}_{-\infty} = 0, P_{-\infty} = \Pi$;
3: for $k = -\infty : 0$ do
4:    // Local data reaches steady state.
5:    For $i = 1 : N$ do
6:        $P_{i,k} = [(AP_{i,k-1}A' + Q)^{-1} + C_i'R_i^{-1}C_i]^{-1}$ ;
7:        $\hat{x}_{i,k} = A\hat{x}_{i,k-1} + P_{i,k}C_i'R^{-1}(y_{i,k} - C_iA\hat{x}_{i,k-1})$ ;
8:    end for
9:    // The remote estimator reaches steady state.
10:   $P_k = [(AP_{k-1}A' + Q)^{-1} + C'R^{-1}C]^{-1}$ ;
11:   $\hat{x}_k = A\hat{x}_{k-1} + P_kC'R^{-1}(y_k - CA\hat{x}_{k-1})$ ;
12: end for
13: // GMM clustering by the EM algorithm.
14: Set $\bar{P}_i = P_{i,0}, \Sigma_i^{(1)} = P_{i,0}$
15: for $k = 1 : +\infty$ do
16:    for $i = 1 : N$ do
17:        $\hat{x}_{i,k} = A\hat{x}_{i,k-1} + \bar{P}_iC_i'R_i^{-1}(y_{i,k} - C_iA\hat{x}_{i,k-1})$;
18:    end for
19:    // the EM algorithm.
20:    Initialize $\pi_k^{(1)}, \pi_k^{(2)}, \mu_k^{(1)}, \mu_k^{(2)}, \Sigma_k^{(2)}$ ;
21:    while $\mathscr{L}(\Phi_{k}; \mathscr{Z}_k)$ not achieve the maximum likelihood estimates do
22:        The expectation step: calculate $\gamma_{i,k}^{(1)}$ and $\gamma_{i,k}^{(2)}$ according to Equation (15)-(16).
23:        The maximization step: calculate $\{\pi_k^{(q)}, \mu_k^{(q)}, \Sigma_k^{(2)}\}_{q=1}^2$ by Equation (17)-(19).
24:    end while
25:    // the error compensator.
26:    $\bar{\gamma}_k^{(2)} = \sum_{i=1}^N \gamma_{i,k}^{(2)}/N$
27:    for $i = 1 : N$ do
28:        if $\bar{\gamma}_k^{(2)} > \varepsilon$ $(\varepsilon \geqslant (|s|/N))$ then
29:            $\hat{\gamma}_{i,k}^{(1)} = \gamma_{i,k}^{(2)}$;
30:            $\hat{\gamma}_{i,k}^{(2)} = \gamma_{i,k}^{(1)}$;
31:        else
32:            $\hat{\gamma}_{i,k}^{(1)} = \gamma_{i,k}^{(1)}$;
33:            $\hat{\gamma}_{i,k}^{(2)} = \gamma_{i,k}^{(2)}$;
34:        end if
35:    end for
36:    // Remote state estimation.
37:    $\hat{x}_k^- = A\hat{x}_{k-1}$ ;
38:    $P_k^- = AP_{k-1}A' + Q$ ;
39:    $\hat{x}_k = \hat{x}_k^- + \sum_{i=1}^N \hat{\gamma}_{i,k}^{(1)}P_kC_i'R_i^{-1}(y_{i,k} - C_i\hat{x}_k^-)$ ;
40:    $P_k = [(P_k^-)^{-1} + \sum_{i=1}^N \gamma\wedge_{i,k}^{(1)}C_i'R_i^{-1}C_i]^{-1}$ ;
41: end for
```

ALGORITHM 1: The GMM-based state estimation against dynamic attacks.

where $\hat{\gamma}_{i,k}^{(1)}$ and $\hat{\gamma}_{i,k}^{(2)}$ are the modified beliefs, and $\varepsilon \geq s/N$ represents a threshold, which can be adjusted according to the performance requirements of the actual system.

3.3. The GMM-Based State Estimation Approach against Dynamic Attacks.
In this subsection, a GMM-based estimation algorithm is proposed to deal with the dynamic attacks, which can improve the estimation accuracy effectively.

$$\hat{x}_k^- = A\hat{x}_{k-1}, \tag{22a}$$

$$P_k^- = AP_{k-1}A' + Q, \tag{22b}$$

$$\hat{x}_k = \hat{x}_k^- + \sum_{i=1}^N \hat{\gamma}_{i,k}^{(1)} P_k C_i' R_i^{-1}(y_{i,k} - C_i\hat{x}_k^-), \tag{22c}$$

$$P_k = \left[(P_k^-)^{-1} + \sum_{i=1}^N \gamma\wedge_{i,k}^{(1)} C_i'R_i^{-1}C_i\right]^{-1}, \tag{22d}$$

where the initial values $\hat{x}_0$ and P_0 are the steady-state values of the remote estimator when $k \leq 0$.
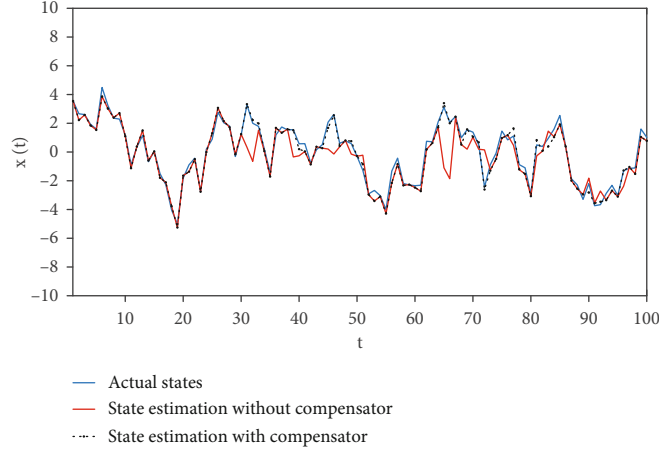
FIGURE 2: The actual states and its remote estimation with different approaches.

Theorem 2. *Consider the linear time-invariant system (1)-(2) and the dynamic adversary satisfying Assumptions 1–4, and the remote state estimation $\widehat{x}_k$ can be calculated by*

Proof. According to the Definition 2 in [16, 28], if $s(s \leq N/2)$ sensors are attacked, the following system is still observable in the absence of attacks:

$$
\begin{aligned}
x_{k+1} &= Ax_k + w_k, \\
y_{\bar{s},k} &= C_{\bar{s}} x_k + v_{\bar{s},k},
\end{aligned}
\tag{23}
$$

where $\bar{s} \subseteq \{1, 2, \cdots, N\}$ is the set of unattacked sensors, and $y_{\bar{s},k}$ is the measurement stacked by the set $\bar{s}$. Similarly, $C_{\bar{s}}$ and $v_{\bar{s},k}$ are the system parameter and the measurement noise stacked by the set $\bar{s}$, respectively. The pair $(A, C_{\bar{s}})$ is observable.

According to Section II, Equation (6) can be expanded as

$$
\begin{aligned}
\widehat{x}_k &= = \widehat{x}_k^- + P_k C' R^{-1} (y_k - C\widehat{x}_k^-) \\
&= \widehat{x}_k^- + P_k \begin{bmatrix} C_1' \\ \vdots \\ C_N' \end{bmatrix}' \begin{bmatrix} R_1 & 0 & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & R_N \end{bmatrix}^{-1} \left(\begin{bmatrix} y_{1,k} \\ \vdots \\ y_{N,k} \end{bmatrix} - \begin{bmatrix} C_1' \\ \vdots \\ C_N' \end{bmatrix} \widehat{x}_k^- \right) \\
&= \widehat{x}_k^- + P_k \begin{bmatrix} C_1' R_1^{-1} \\ \vdots \\ C_N' R_N^{-1} \end{bmatrix}' \begin{bmatrix} y_{1,k} - C_1' \widehat{x}_k^- \\ \vdots \\ y_{N,k} - C_N' \widehat{x}_k^- \end{bmatrix} \\
&= \widehat{x}_k^- + \sum_{i=1}^{N} P_k C_i' R_i^{-1} (y_{i,k} - C_i \widehat{x}_k^-),
\end{aligned}
\tag{24}
$$

where the default weight of each sensor is equal to 1 when the sensor is not attacked. □

Based on the above analysis, we can calculate the remote state estimation $\widehat{x}_k$ by adopting the undamaged sensors. The belief $\widehat{\gamma}_{i,k}^{(1)}$ represents the probability that the sensor i is

undamaged. Then, we can fuse the local data by adopting $\widehat{\gamma}_{i,k}^{(1)}$ as the new weight of the local data, and then the Equations (22a)-(22d) can be obtained.

The system is assumed to reach steady state before time $k = 0$. The adversary can launch dynamic attacks at any time when $k \geq 1$. Starting from time $k = 1$, the local state estimation $\widehat{x}_{i,k}$ is calculated utilizing the measurement of sensor i at each time instant k. Based on that, the remote estimator clusters the local state estimates and calculates the parameter Φ_k by the EM algorithm according to Equation (15)-(19). Then, the error compensator is used to correct the error beliefs. Finally, based on the modified belief $\widehat{\gamma}_{i,k}^{(1)}$, the local data can be fused by Theorem 2 to obtain the state estimation $\widehat{x}_k$. The whole process is summarized in Algorithm 1.

4. Numerical Simulation

In this section, the effectiveness of the GMM-based estimation algorithm is verified through numerical simulations. Similar to literature [21], we consider a linear time-invariant dynamic process which is measured by 15 sensors. The system parameters A and Q are randomly generated from intervals [0.4, 0.99] and [0.5, 2], respectively. Matrices C_i and R_i, $i \in N$, are randomly generated from intervals [1, 2]. The system reaches steady state before $t = 30$, and the attack signal starts from time $t = 31$, assuming that s $(1 \leq s \leq 6)$ sensors are attacked by $a_{i,k}$ at each time instant $t (t \geq 31)$.

4.1. Example 1. In this example, the estimation accuracy of GMM-based method with and without compensator against dynamic attacks has been compared. Similar to [15], the attack signal $a_{i,k}$ can be assumed to be a linear function of the measurement noise:

$$
a_{i,k} = \beta v_{i,k} + \Theta,
\tag{25}
$$

where β and Θ are real number from the interval [-5, 5] and [-10, 10], respectively. Meanwhile, $a_{i,k}$ satisfies Assumptions 1–4.
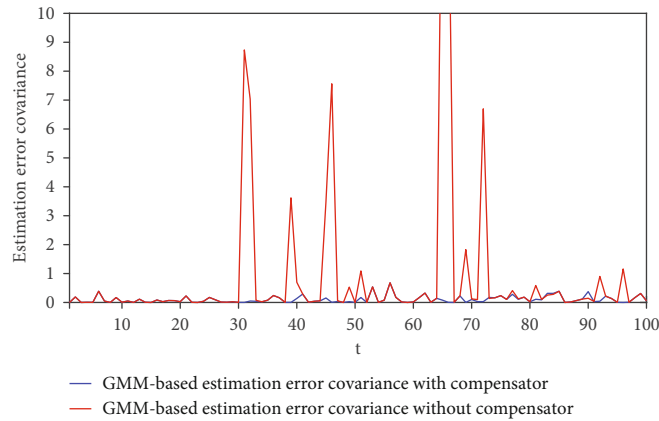
FIGURE 3: Remote estimation error covariance of the GMM-based method with and without compensator.
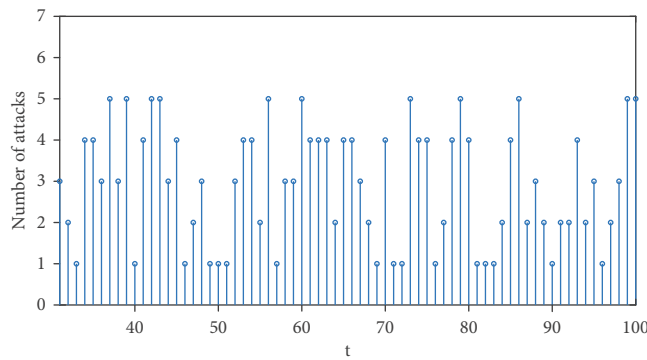
FIGURE 4: The number of attacked sensors at each moment when $T \geqslant 31$.
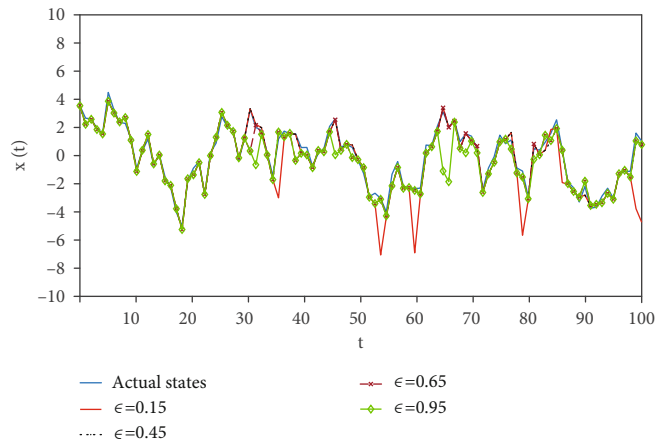
FIGURE 5: Comparison of the state estimation with different thresholds.

Set the threshold $\varepsilon = 0.45$ in the following example. In Figure 2, the trajectories of the actual state and the states estimated by the GMM-based estimation method with and without compensator are plotted. It is shown that the estimated states of the GMM-based method with compensator (dotted line) are closer to the actual state than that without compensator (red line). Figure 3 shows the estimation error covariance for the GMM-based method with and without compensator, respectively. It

is observed that the estimation error covariance of the method without compensator (red line) is larger than that with compensator (black line), which means that the error compensator proposed in this paper can effectively reduce the impact of faulty data on state estimation. According to Figures 2 and 3, the estimation accuracy of the GMM-based estimation method with the compensator is higher than that without the compensator against dynamic attacks.
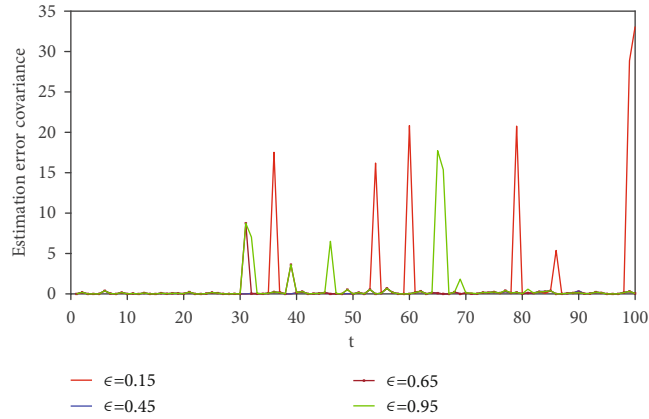
Figure 6: Remote estimation error covariance with different thresholds.
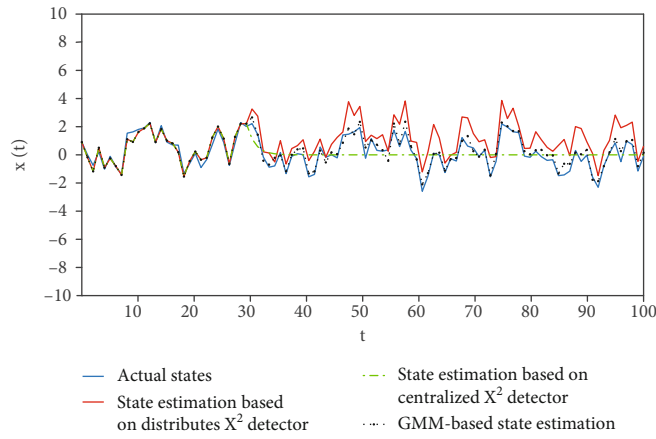
Figure 7: The actual states and its remote estimation based on different methods.
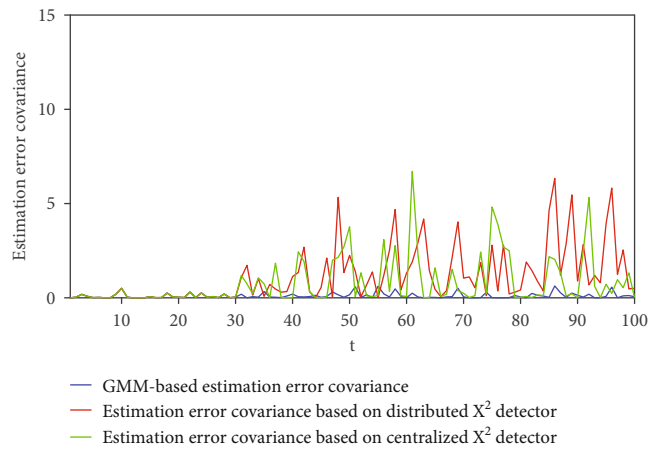
Figure 8: Remote estimation error covariance based on different methods.

The number of attacked sensors at each moment when $T \geq 31$ is plotted in Figure 4, and the state estimation and corresponding error covariance of the GMM-based algorithm when the compensator takes different thresholds are shown in Figures 5 and 6, respectively. It is seen that the state estimation accuracy is higher when $\varepsilon = 0.45$ and 0.65 than $\varepsilon = 0.15$ and 0.95, which is indicated that the performance of the remote estimator will deteriorate while ε is too large or too small. Hence, the threshold can be adjusted according to the actual performance requirements of the real system.

4.2. Example 2. Distributed and centralized χ^2 false-data detectors are common, and they determine whether an attack exists based on the statistical characteristics of the innovation $y_{i,k} - C_i\widehat{x}_{i,k}^-$ and $y_k - C\widehat{x}_k^-$, respectively. From [21], a well-designed dynamic attack can successfully bypass the distributed χ^2 detector but fails to remain stealthy to the centralized χ^2 false-data detector. In this subsection, we have compared the proposed approach and the estimation methods based on different χ^2 false-data detectors.

Similar to [21], the attack signal $a_{i,k}$ is set as

$$a_{i,k} = -2y_{i,k} + 2C_i\widehat{x}_k^-, \tag{26}$$

where $a_{i,k}$ satisfies Assumptions 1–4.

In Figure 7, the trajectories of the actual state and the state estimated by estimation methods based on different detectors are plotted, respectively. It is seen that the GMM-based state estimation (black line) is closer to the actual state than the state estimation based on the distributed and centralized χ^2 detector (red and green). Figure 8 shows the estimation error covariance of the corresponding methods, and it is observed that the GMM-based estimation error covariance is much smaller than that based on the distributed and centralized χ^2 detector. It can be seen that the GMM-based estimation approach proposed in this paper can improve the performance effectively.

5. Conclusion

This paper studies the state estimation problem against dynamic malicious attacks. An error compensator is presented, which can reduce the influence of local error data on state estimation effectively. Based on that, a new GMM-based state estimation algorithm is proposed to improve the estimation accuracy for the system suffering from dynamic attacks. Finally, the effectiveness of the proposed algorithm is verified by numerical simulations. We will extend the GMM-based approach further to systems with parametric uncertainties in the future.

Data Availability

Some or all data, models, or code generated or used during the study are available from the corresponding author by request (Cui Zhu).

Conflicts of Interest

The authors declare that they have no conflicts of interest related to this work.

Acknowledgments

References

[1] K.-J. Park, R. Zheng, and X. Liu, "Cyber-physical systems: milestones and research challenges," *Computer Communications*, vol. 36, no. 1, pp. 1–7, 2012.

[2] R. Langner, "Stuxnet: dissecting a cyberwarfare weapon," *IEEE Security and Privacy*, vol. 9, no. 3, pp. 49–51, 2011.

[3] J. P. Farwell and R. Rohozinski, "Stuxnet and the future of cyber war," *Survival*, vol. 53, no. 1, pp. 23–40, 2011.

[4] J. P. Conti, "The day the samba stopped [power blackouts]," *Engineering and Technology*, vol. 5, no. 4, pp. 46-47, 2010.

[5] J. Slay and M. Miller, "Lessons learned from the Maroochy water breach," *IFIP Advances in Information and Communication Technology*, vol. 253, pp. 73–82, 2007.

[6] K. Kim and P. R. Kumar, "Cyber–physical systems: a perspective at the centennial," *Proceedings of the IEEE*, vol. 100, pp. 1287–1308, 2012.

[7] L. Shan, J. Yu, J. Zhang, Y. Li, E. Zhou, and L. Zhao, "Distributed state estimation based on the realtime dispatch and control cloud platform," in *2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2)*, pp. 1–6, Beijing, China, 2018.

[8] Y. Kobayashi and I. Ono, "Sequential estimation of states and parameters of nonlinear state space models using particle filter and natural evolution strategy," in *2020 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1–8, Glasgow, UK, 2020.

[9] I. Polyakov, A. Pazderin, and O. Polyakova, "Computational performance comparing of the state estimation problem statementes in polar and rectangular coordinates," in *2019 16th Conference on Electrical Machines, Drives and Power Systems (ELMA)*, pp. 1–4, Varna, Bulgaria, June 2019.

[10] M. M. Rana and N. Dahotre, "Bayesian network and semidefinite programming based wireless power transfer manufacturing system state estimation and regulation," in *2021 23rd International Conference on Advanced Communication Technology (ICACT)*, pp. 237–241, PyeongChang, Korea (South), Feb. 2021.

[11] J. Zhao, G. Zhang, Z. Y. Dong, and M. La Scala, "Robust forecasting aided power system state estimation considering state correlations," *IEEE Transactions on Smart Grid*, vol. 9, no. 4, pp. 2658–2666, 2018.

[12] M. M. Rana, W. Xiang, and E. Wang, "IoT-based state estimation for microgrids," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 1345-1346, 2018.

[13] M. Netto and L. Mili, "A robust data-driven Koopman Kalman filter for power systems dynamic state estimation," *IEEE Transactions on Power Systems*, vol. 33, no. 6, pp. 7228–7237, 2018.

[14] J. Zhao, Z. Zheng, S. Wang et al., "Correlation-aided robust decentralized dynamic state estimation of power systems with unknown control inputs," *IEEE Transactions on Power Systems*, vol. 35, no. 3, pp. 2443–2451, 2020.

[15] S. Mishra, Y. Shoukry, N. Karamchandani, S. N. Diggavi, and P. Tabuada, "Secure state estimation against sensor attacks in the presence of noise," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 1, pp. 49–59, 2017.

[16] W. Ao, Y. Song, C. Wen, and J. Lai, "Finite time attack detection and supervised secure state estimation for CPSs with malicious adversaries," *Information Sciences*, vol. 451–452, pp. 67–82, 2018.

[17] X. Liu, Y. Mo, and E. Garone, "Secure dynamic state estimation by decomposing Kalman filter," *IFAC (International Federation of Automatic Control)*, vol. 50, no. 1, pp. 7351–7356, 2017.

[18] X. Liu, Y. Mo, and X. Ren, "Security analysis of continuous-time cyber-physical system against sensor attacks," in *2017 13th IEEE Conference on Automation Science and Engineering (CASE)*, pp. 1586–1591, Xi'an, China, Aug. 2017.

[19] N. Forti, G. Battistelli, L. Chisci, and B. Sinopoli, "Secure state estimation of cyber-physical systems under switching attacks," *IFAC PapersOnLine*, vol. 50, no. 1, pp. 4979–4986, 2017.

[20] A. Y. Lu and G. H. Yang, "Secure state estimation for cyber-physical systems under sparse sensor attacks via a switched Luenberger observer," *Information Sciences*, vol. 417, pp. 454–464, 2017.

[21] Z. Guo, D. Shi, D. E. Quevedo, and L. Shi, "Secure State Estimation against Integrity Attacks: a Gaussian Mixture Model Approach," *IEEE Transactions on Signal Processing*, vol. 67, no. 1, pp. 194–207, 2019.

[22] F. Wen and Z. Wang, "Distributed Kalman filtering for robust state estimation over wireless sensor networks under malicious cyber attacks," *Digital Signal Processing*, vol. 78, pp. 92–97, 2018.

[23] H. Fawzi, P. Tabuada, and S. Diggavi, "Secure estimation and control for cyber-physical systems under adversarial attacks," *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1454–1467, 2014.

[24] M. Pajic, J. Weimer, N. Bezzo et al., "Robustness of attack-resilient state estimators," in *2014 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS)*, pp. 163–174, Berlin, Germany, April 2014.

[25] B. D. Anderson and J. B. Moore, *Optimal Filtering*, Courier Corporation, 2012.

[26] C. M. Bishop, *Pattern Eecognition and Machine Learning*, Springer, 2006.

[27] T. K. Moon, "The expectation-maximization algorithm," *IEEE Signal Processing Magazine*, vol. 13, no. 6, pp. 47–60, 1996.

[28] M. S. Chong, M. Wakaiki, and J. P. Hespanha, "Observability of linear systems under adversarial attacks," in *Proceedings of the American Control Conference*, pp. 2439–2444, Chicago, IL, USA, 2015.

Research Article

Adaptive Robust Control for Uncertain Systems via Data-Driven Learning

Jun Zhao [1] and Qingliang Zeng [1,2]

¹College of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Qingdao 266590, China
²Department of Information Science and Engineering, Shandong Normal University, Jinan 250358, China

Correspondence should be addressed to Qingliang Zeng; qlzeng@sdust.edu.cn

Although solving the robust control problem with offline manner has been studied, it is not easy to solve it using the online method, especially for uncertain systems. In this paper, a novel approach based on an online data-driven learning is suggested to address the robust control problem for uncertain systems. To this end, the robust control problem of uncertain systems is first transformed into an optimal problem of the nominal systems via selecting an appropriate value function that denotes the uncertainties, regulation, and control. Then, a data-driven learning framework is constructed, where Kronecker's products and vectorization operations are used to reformulate the derived algebraic Riccati equation (ARE). To obtain the solution of this ARE, an adaptive learning law is designed; this helps to retain the convergence of the estimated solutions. The closed-loop system stability and convergence have been proved. Finally, simulations are given to illustrate the effectiveness of the method.

1. Introduction

Existing achievements of control techniques are mostly acquired under the assumption that there are no dynamical uncertainties in the controlled plants. Nevertheless, in practical control systems, there are many external disturbances and/or model uncertainties, so the system lifetimes are always affected by those uncertainties. The factors of uncertainties must be taken into consideration in the design of the controller such that the closed-loop systems must have good responses even in the presence of such uncertain dynamics. We say a controller is robust if it works even though the practical system deviates from its nominal model. Therefore, it creates the problem of robust control design, which has been widely studied during the past decades [1, 2]. The latest research [1, 3] shows that the robust control problem can be addressed via using the optimal control approach for the nominal system. Nevertheless, the online

solution for the derived optimal control problem is not handled in [1].

Considering optimal control problems, recently, many approaches have been presented [4, 5]. A linear system optimal control problem is described to address the associated linear quadratic regulator (LQR) problem, where the optimal control law can be obtained. The theory of dynamic programming (DP) has been proposed to study the optimal control problem in the past years [6]; however, there is an obvious disadvantage for DP, i.e., with the increase in the dimensions of system state and control input, there is an alarming increase in the amount of computation and storage, which is called "curse of dimensionality." To overcome this problem, the neural network (NN) is used to approximate the optimal control problem [7], which leads to recent research work on adaptive/approximate dynamic programming (ADP); the tricky optimal problem can be tackled via ADP method; thus, we can get the online solution of the

optimal cost function [8]. Recently, robust control design based on adaptive critic idea has gradually become one of the research hotspots in the field of ADP. Many methods have been proposed one after another, which are collectively referred to as the robust adaptive critic control. A basic approach is to transform the problem to establish a close relationship between robustness and optimality [9]. In these literatures, the closed-loop system generally satisfies the uniformly ultimately bounded (UUB). These results fully show that the ADP method is suitable for the robust control design of complex systems in uncertain environment. Since many previous ADP results are not focus on the robust performance of the controller, the emergence of robust adaptive critic control greatly expands the application scope of ADP methods. Then, considering the commonness in dealing with system uncertainties, the self-learning optimization method combined with ADP and sliding mode control technology provides a new research direction for robust adaptive critic control [10]. In addition, the robust ADP method is another important achievement in this field. It is worth mentioning that the application of robust ADP methods in power systems has attracted special attention [11], leading to a higher application value in industrial systems.

Based on the above facts, we develop a robust control design for uncertain systems via using an online data-driven learning method. For this purpose, the robust control problem of uncertain systems is first transformed into an optimal control problem of the nominal systems with an appropriate cost function. Then, a data-driven technique is developed, where Kronecker's products and vectorization operations are used to reformulate the derived ARE. To solve this ARE, a novel adaptive law is designed, where the online solution of ARE can be approximated. Simulations are given to indicate the validity of the developed method.

The major contributions of this paper include the following:

(1) To address the robust control problem, we transform the robust control problem of uncertain systems into an optimal control problem of the nominal system. It provides an approach to address the robust control problem

(2) Kronecker's products and vectorization operations are used to reformulate the derived ARE, which can help to rewrite the original ARE into a linear parametric form. It gives a new pathway to online solve the ARE

(3) A newly developed adaptation algorithm driven by the parameter estimation errors is used to online learn the unknown parameters. The convergence of the estimated unknown parameters to the true values can be guaranteed

This paper is organized as follows: In Section 2, we introduce the robust control problem and transform the robust control problem into an optimal control problem. In Section 3, we design an ADP-based data-driven learning method to online solve the derived ARE, where Kronecker's products

and vectorization operations are used. Section 4 gives some simulation results to illustrate the effectiveness of the proposed method. Some conclusions are stated in Section 5.

2. Preliminaries and Problem Formulation

A continuous-time (CT) uncertain system can be written as

$$\dot{x}(t) = A(d)x + Bu + D(x), \qquad (1)$$

where $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$ are the system state and the control action, respectively. $A \in \mathbb{R}^{n \times n}$ is the system matrix and $B \in \mathbb{R}^{n \times m}$ is the input matrix. $d \in \Omega$ denotes the uncertain parameter involved in the system, and $D(x)$ denotes the bounded nonlinearities. The purpose of this paper is designing a controller to make the system (1) asymptotically stable under uncertainties $d \in \Omega$.

In this paper, we study the case, i.e., the matching condition is satisfied; in other words, the uncertainty is in the range of B; thus, the uncertainty is in matrix A which can be rewritten as $A(d) - A(d_0) = B\omega(d)$ for uncertain $\omega(d)$, where $d_0 \in \Omega$ is the nominal value of d. Let F denote the upper bound of $\omega(d)$; then, for all $d \in \Omega$, we have $\omega^T(d)\omega(d) \le F$. In this paper, we will resolve following problem, i.e., realize the online solution for robust control with uncertain system (1). Then, the above robust control problem can be rewritten as

$$\dot{x}(t) = A(d_0)x + Bu + B\omega(d)x. \qquad (2)$$

To obtain the robust control solution, the classical method is linear matrix inequality (LMI) [12] in an *offline*; **online** resolving the robust control problem is not easy. To overcome this problem, the authors in [1, 9] reported that the robust control problem of uncertain systems can be transformed into an optimal control problem of nominal systems, which provides a new pathway to address the robust control problem. Hence, consider the nominal plant of the system (1).

$$\dot{x}(t) = A(d_0)x + Bu. \qquad (3)$$

The aim is to find a control action u to minimize the following continuous cost function:

$$V(x) = \int_t^\infty \left(x^T Fx + x^T Qx + u^T Ru\right)d\iota, \qquad (4)$$

where $Q = Q^T > 0 \in \mathbb{R}^{n \times n}$ and $R = R^T > 0 \in \mathbb{R}^{m \times m}$ are the weight matrices.

It should also be noted that the upper bound F of the uncertainties $\omega(d)$ is involved in the cost function (4) to address their effects. The following Lemma summarizes the equivalence between the robust control of the system (1) or (2) and the optimal control of the system (3) with cost function (4).

Lemma 1 (see [9]). *If the solution to the optimal control problem of the nominal system (3) with cost function (4)*

exists, then it is a solution to the robust control problem for system.

Lemma 1 exploits the relationship between the robust control and optimal control and thus provides a new way to address the robust control.

To address the optimal control problem of (3), an Algebraic Riccati equation (ARE) can be derived via the cost function (4) as

$$A^T P^* + P^* A + Q_T - P^* B R^{-1} B^T P^* = 0, \qquad (5)$$

where P^* is the solution of (5), $Q_T = Q + F$, and $A = A(d_0)$. Then, based on the optimal principle, its optimal action can be given as

$$u = -Kx = -R^{-1} B^T P^* x. \qquad (6)$$

3. Online Solution to Robust Control via Data-Driven Learning

This section will propose a data-driven learning method to resolve the robust control, the schematic of the proposed control method as given Figure 1.

To this end, the system states are multiplied on both sides of ARE (5); we have

$$x^T \left(A^T P^* + P^* A - P^* B R^{-1} B^T P^* \right) x = -x^T Q_T x. \qquad (7)$$

We apply two operations ($\mathrm{vec}(\cdot)$ and $\otimes$) on (7) yielding

$$2(x \otimes Ax)^T \mathrm{vec}(P^*) + (x \otimes x)^T \mathrm{vec}(Q_T) - \left(\mathrm{vec}(B R^{-1} B^T) \otimes (x \otimes x) \right)^T \mathrm{vec}(P^* \otimes P^*) = 0. \qquad (8)$$

Since the $\mathrm{vec}(P^* \otimes P^*)$ is involved in (8), then the dimension of (8) is very high. To overcome this issue, a dimensionality reduction operation on (7) is given

$$x^T \left(A^T P^* + P^* A - K^T R^{-1} K \right) x = -x^T Q_T x, \qquad (9)$$

then we can apply two operations ($\mathrm{vec}(\cdot)$ and $\otimes$) on (9) yielding

$$2(x \otimes Ax)^T \mathrm{vec}(P^*) + (x \otimes x)^T \mathrm{vec}(Q_T) - (\mathrm{vec}(R) \otimes (x \otimes x))^T \mathrm{vec}(K \otimes K) = 0. \qquad (10)$$

Hence, above equation (10) can be rewritten as a compact form

$$\phi = -W^T \vartheta, \qquad (11)$$

where $W = [\mathrm{vec}(P^*), \mathrm{vec}(K \otimes K)]^T$, $\vartheta = [2(x \otimes Ax), -\mathrm{vec}(R) \otimes (x \otimes x)]$, and $\phi = (x \otimes x)\mathrm{vec}(Q_T)$.

3.1. Online Solution of Robust Control. From (11), we have that only variable W is unknown due to involving the unknown matrices P^* and K; thus, the next operation is
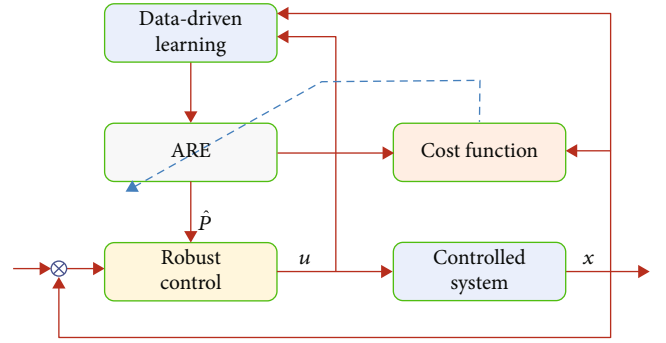
FIGURE 1: Schematic of the proposed control method.

design an online learning method to update the unknown variable W. Consequently, the unknown matrices P^* and K can be online estimated based on the estimate $\widehat{W}$ of W. To this end, we define two auxiliary variables, i.e., $Y \in \mathbb{R}^{l \times l}$ and $N \in \mathbb{R}^l$ as

$$\begin{cases} \dot{Y} = -\ell Y + \vartheta \vartheta^T, \ Y(0) = 0, \\ \dot{N} = -\ell N + \vartheta \phi^T, \ N(0) = 0, \end{cases} \qquad (12)$$

with $\ell > 0$ being the learning parameter. Then, its solution can be calculated as

$$\begin{cases} Y = \int_0^t e^{-\ell(t-r)} \vartheta(\iota) \vartheta^T(\iota) d\iota, \\ N = \int_0^t e^{-\ell(t-r)} \vartheta(\iota) \phi^T(\iota) d\iota. \end{cases} \qquad (13)$$

To realize the online estimation for $\widehat{W}$ based on the estimation error $\tilde{W}$, an auxiliary vector $M \in \mathbb{R}^l$ is defined as

$$M = Y\widehat{W} + N. \qquad (14)$$

After taking (11) into (13), we have $N = -YW$; thus, we can rewrite (14) as

$$M = Y\widehat{W} + N = -Y\tilde{W}, \qquad (15)$$

with $\tilde{W} = W - \widehat{W}$ being the estimation error. Then, we can design the adaptive learning law as

$$\dot{\widehat{W}} = -\kappa M, \qquad (16)$$

with κ being the learning gain.

For adaptive law (16), auxiliary vector M of (14) obtained based on and ϑ using (15) contains the information on the parameter estimation error $Y\tilde{W}$. Thus, M can be used to drive parameter estimation. Consequently, parameter estimation $\widehat{W}$ can be updated along with the estimation error $\tilde{W}$ extracted by using the measurable system states x. Thus, this adaptive algorithm clearly differs to the gradient descent algorithms used in other ADP literatures.

1) (Initialization): given the initial parameter $\widehat{W}(0)$ and gains κ, ℓ for adaptive learning law (16)
2) (Measurement): measure the system input\output data and construct the regressors ϕ, ϑ in (10) and (11)
3) (Online adaptation): solve Y, N, and M and learn the unknown parameter $\widehat{W}$ with (16) to obtain the control u
4) (Apply control): apply the derived output-feedback control u on the system

ALGORITHM 1: (Step-by-step Implementation for Online Robust Control Solution of Uncertain Systems).

Since the fact $M = -Y\widetilde{W}$ is true, then we can obtain the following lemma as follows.

Lemma 2 (see [13, 14]. *Assume that the variable ϑ provided in (12) meets persistently excited, then the matrix Y given in (12) can be considered as positive definite, which means that $\lambda_{\min}(Y) > \sigma > 0$ for any positive constant σ.*

Lemma 2 shows the positivity of the variable ϑ, then we can summarize the convergence of proposed adaptive learning law (16) as follows.

Theorem 3. *Consider (11) with adaptive learning law (16), when variable ϑ provided in (11) satisfies PE condition, then the estimation error $\widetilde{W}$ is convergence to the origin.*

Proof. A Lyapunov function can be chosen as $V_1 = (1/2)(\widetilde{W}^T \kappa^{-1} \widetilde{W})$, then we can calculate its $\dot{V}_1$ as

$$\dot{V}_1 = \widetilde{W}^T \kappa \widetilde{W} = -\widetilde{W}^T Y \widetilde{W} \leq -\sigma \|\widetilde{W}\|^2 \leq -\mu V_1, \qquad (17)$$

with $\mu = 2\sigma/\lambda_{\max}(\kappa^{-1}) > 0$. Hence, we have the estimated error $\widetilde{W} \longrightarrow 0$. This completes the proof.$\square$

The step-by-step implementation of proposed learning algorithm is given as follows.

Remark 4. For the above designed adaptive learning law (16), which is derived by the estimation error. To this end, the control input u and system states x are used; this is clearly different to the existing results [15]. In particular, two operations vec$(\cdot)$ and $\otimes$ are applied to the derived ARE; this helps to realize the online learning. Consequently, faster convergence can be retained compared to the previous gradient method-based adaptive laws designed.

Remark 5. It is a fact that some ADP methods are applied to address the robust control problem successfully. However, most existing ADP techniques focus on *H-infinity* control problem. For proposed robust control problem in this paper, we know the uncertain parameter d are involved in system matrix A such that $A(d)$, so we can consider the system contains unmolded dynamics. To obtain the uncertain term bound, we should do some operations such that $A(d) - A(d_0)$, which will be used in the cost function (4). Assume that the system dynamics are completely unknown, the uncertain bound may not be used in cost function as expected. Hence, the system matrix must be known in this paper; future work

will try to solve the output-feedback robust control under completely unknown dynamics.

3.2. Stability Analysis. Before the stability analysis of the closed-loop system, we first define the practical optimal control as

$$u = -R^{-1}B^T \widehat{P}x, \qquad (18)$$

with $\widehat{P}$ being the estimated P.

Taking (18) into (3), we have the closed-loop system dynamics

$$\dot{x} = Ax + Bu = (A - BR^{-1}B^T \widehat{P})x. \qquad (19)$$

To complete the stability analysis, we use the following assumptions as follows.

Assumption 6. The dynamic matrices $A \leq b_A$ and $B \leq b_B$ for $b_A, b_B > 0$, the estimated matrix $\widehat{P} \leq b_P$ for $b_P > 0$.

In fact, the above assumptions are not stringent in practical systems and have been widely used in many results [13, 14, 16].

Now, some results can be included as follows.

Theorem 7. *Consider the system (3) with adaptive learning law (16), if the variable ϑ is PE, then the parameter estimation error $\widetilde{W}$ converges to zero, and the derived control is convergence to its optimal control, i.e., $\|u - u^*\| \longrightarrow 0$.*

Proof. Consider a Lyapunov function as

$$J = J_1 + J_2 = \frac{1}{2}\left(\widetilde{W}^T \kappa^{-1} \widetilde{W}\right) + \Gamma_1 x^T x + K_1 V^*, \qquad (20)$$

where V^* is the optimal cost function provided in (4) and $\Gamma_1 > 0$ and $K_1 > 0$ are the positive constants.

From (17), we have $\dot{J}_1$ as

$$\dot{J}_1 = -\widetilde{W}^T Y \widetilde{W} \leq -\sigma \|\widetilde{W}\|^2. \qquad (21)$$
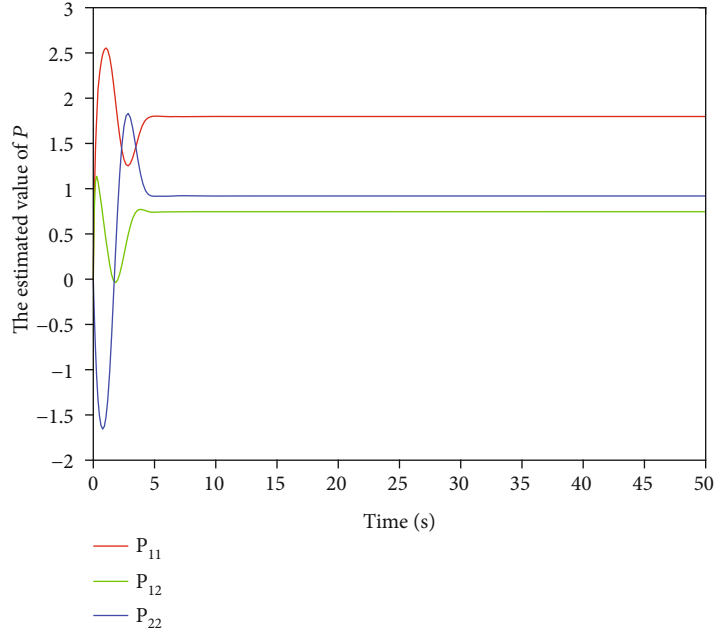
FIGURE 2: Convergence of estimated the matrix P.

Then, the $\dot{J}_2$ can be derived from systems (3) and (19) as

$$
\begin{aligned}
\dot{J}_2 &= 2\Gamma_1 x^T \dot{x} + K_1 \dot{V}^* = 2\Gamma_1 x^T \left(Ax - BR^{-1}B^T \hat{P}\right)x \\
&\quad + K_1 \left(-x^T Q_T x - u^* R u\right) \\
&\leq 2\Gamma_1 \left(b_A - \lambda_{\max}\left(R^{-1}\right) b_B^2 b_P \right. \\
&\quad \left. - K_1 \lambda_{\min}\left(Q_T\right)\right)\|x\|^2 - K_1 \lambda_{\min}(R)\|u^*\|^2.
\end{aligned}
\tag{22}
$$

Thus, based on (21) and (22), we have $\dot{J}$ as

$$
\begin{aligned}
\dot{J} \leq J_1 + J_2 &= -\sigma\left\|\tilde{W}\right\|^2 - 2\Gamma_1\left(b_A - \lambda_{\max}\left(R^{-1}\right) b_B^2 b_P \right. \\
&\quad \left. - K_1 \lambda_{\min}\left(Q_T\right)\right)\|x\|^2 - K_1 \lambda_{\min}(R)\|u^*\|^2.
\end{aligned}
\tag{23}
$$

Then, the parameters Γ_1 and K_1 can be chosen fulfilling following conditions

$$
\begin{aligned}
K_1 &> \max\left\{\frac{1}{\lambda_{\min}}(R), \frac{1}{\lambda_{\min}}(Q_T)\right\}, \\
\Gamma_1 &> \frac{b_A - \lambda_{\max}\left(R^{-1}\right) b_B^2 b_P - K_1 \lambda_{\min}\left(Q_T\right)}{2}.
\end{aligned}
\tag{24}
$$

Therefore, we can rewrite (23) as

$$
\dot{J} \leq -a_1\left\|\tilde{W}\right\|^2 - a_2\|x\|^2 - a_3\|u^*\|^2,
\tag{25}
$$
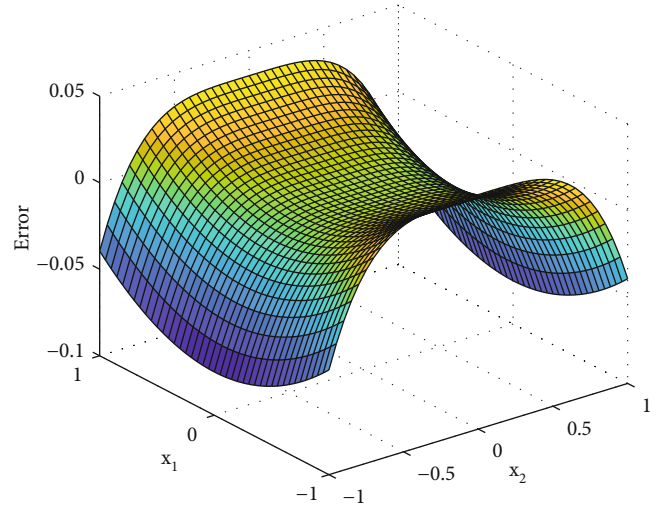
FIGURE 3: The error between $\hat{P}$ and P.

where a_1, a_2, and a_3 are represented as

$$
\begin{aligned}
a_1 &= \sigma, \\
a_2 &= 2\Gamma_1\left(b_A - \lambda_{\max}\left(R^{-1}\right) b_B^2 b_P - K_1 \lambda_{\min}\left(Q_T\right)\right), \\
a_3 &= K_1 \lambda_{\min}(R).
\end{aligned}
\tag{26}
$$

Thus, we have $J \longrightarrow 0$ for $t \longrightarrow \infty$ via Lyapunov theorem, then the estimation error $\tilde{W}$ converges to zero, i.e., $\hat{W} \longrightarrow W$. Consequently, we can obtain the error between u and u^* as

$$
\|u - u^*\| = -R^{-1}B^T \hat{P} - R^{-1}B^T P^* = -R^{-1}B^T\left(\hat{P} - P^*\right) \longrightarrow 0.
\tag{27}
$$
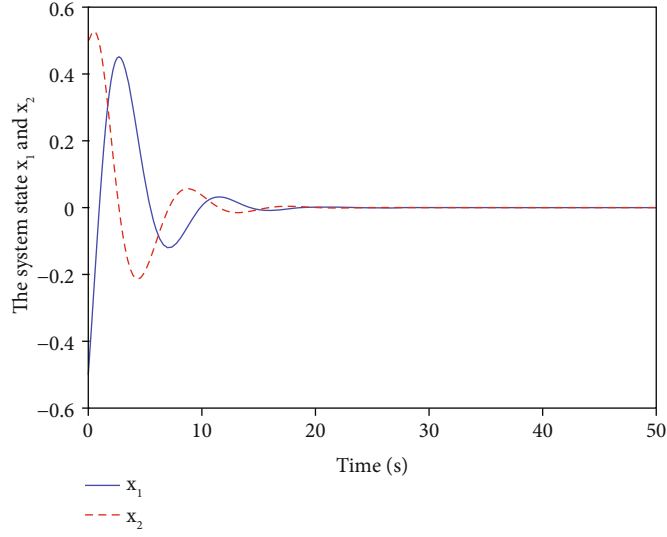
Figure 4: Estimation of system state x.
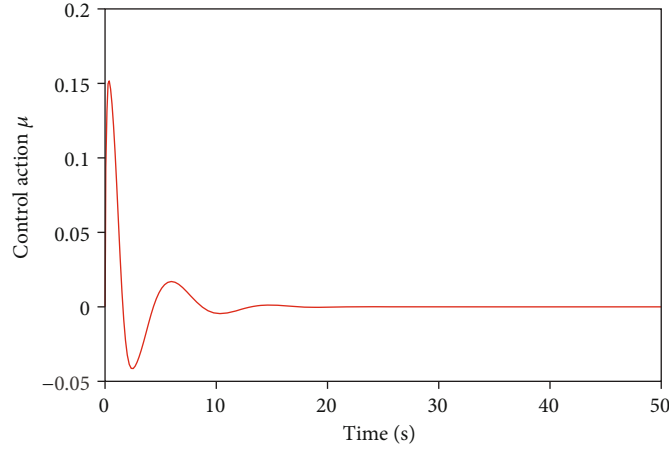
Figure 5: Control action u.

This implies the practical optimal control convergence to 0 is true. This completes the proof.□   □

4. Simulation

4.1. Example 1: Second-Order System. We consider a CT second-order system as

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -0.5 + d & -0.5 + d \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \qquad (28)$$

where $d \in [-0.3, 0.3]$ denotes the uncertainties in system and $x = [x_1, x_2]^T$ is the state variable. The purpose of the paper is to design a control u making the system (28) stable. In this

paper, we define $d_0 = 0$, then based on the stated in Section 2, we can rewrite the system (28) as

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ -0.5 & -0.5 \end{bmatrix} x + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} [d \quad d] \begin{bmatrix} 0 \\ 1 \end{bmatrix} x, \qquad (29)$$

then we can extract the uncertain term as $\omega(d) = [d, d]$. Thus, the upper bound F can be calculated as

$$\omega^T(d)\omega(d) = \begin{bmatrix} d \\ d \end{bmatrix} [d \quad d] \le \begin{bmatrix} 0.3^2 & 0.3^2 \\ 0.3^2 & 0.3^2 \end{bmatrix} = F. \qquad (30)$$

To complete the simulation, we set the initial system states as $x = [0.5, -0.5]^T$, the weights matrices are $Q = I, R = 1$, and learning gains are $\ell = 8.9$ and $\kappa = 96.5$. To show the
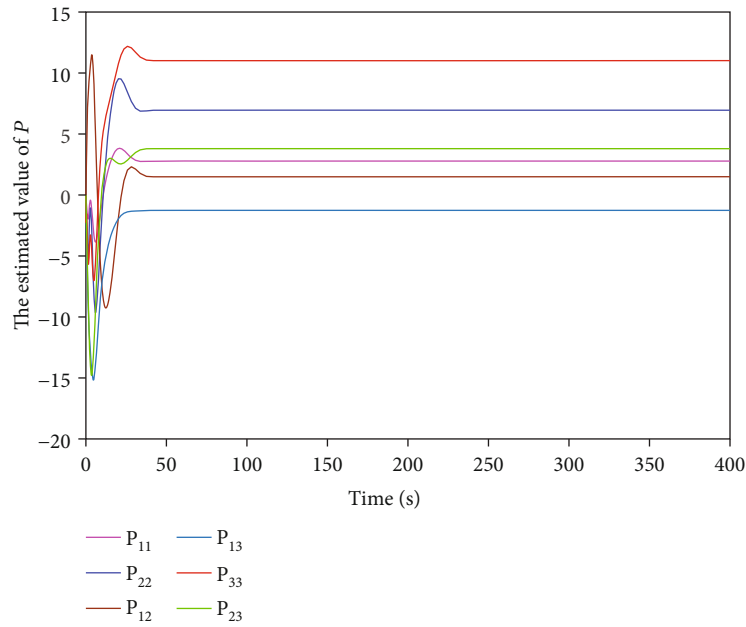
FIGURE 6: Convergence of estimated the matrix P.
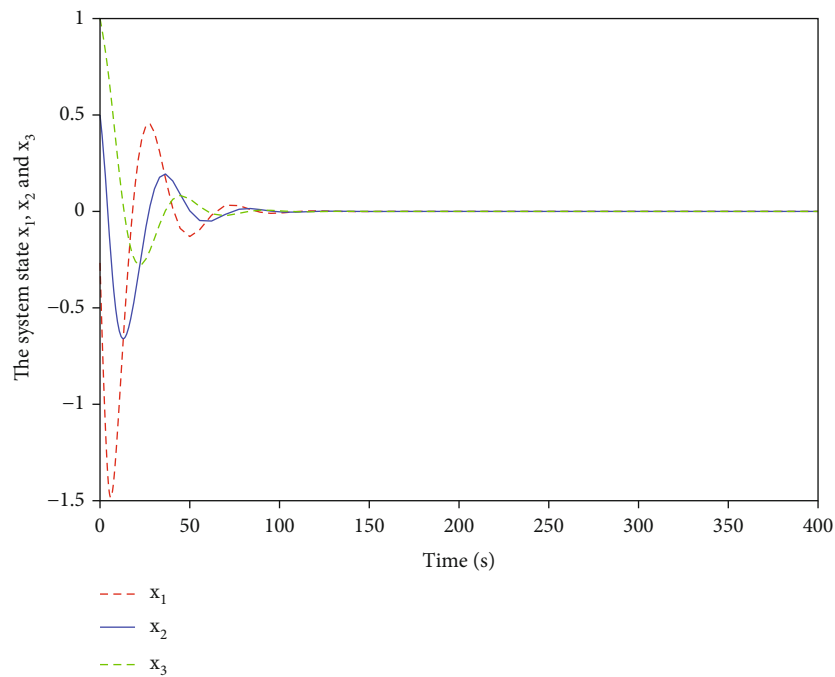
FIGURE 7: Estimation of system state x.

effectiveness of the proposed algorithm, the offline solution of ARE is given as

$$P^* = \begin{bmatrix} 1.5128 & 0.6180 \\ 0.6180 & 1.0767 \end{bmatrix}. \tag{31}$$

Figure 2 gives the estimation of the matrix $\hat{P}$ with online adaptive learning law (16); based on the ideal solution in (31), we have that the estimated solution $\hat{P}$ is convergence to its optimal solution P^*. This is also found in Figure 3, where the normal error, i.e., $\|\hat{P} - P^*\|$, is provided. The good convergence will contribute to the rapid convergence of the system states, which can be found in Figure 4, the system states are bounded and smooth. Since the estimated $\hat{P}$ fast convergence to P^*, then the system response is quite fast; this also can be found in Figure 4. The corresponding control input is given in Figure 5, which is bounded.
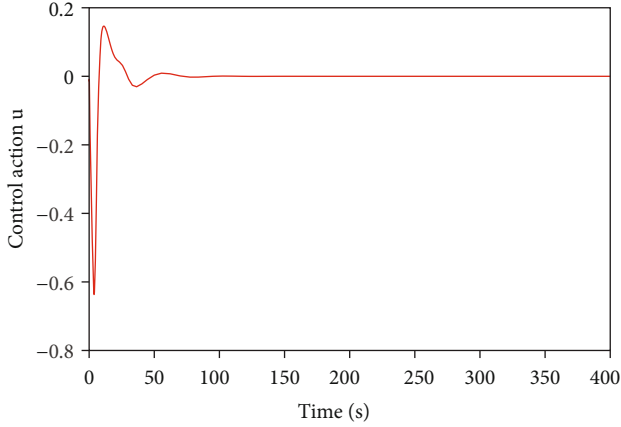
FIGURE 8: Control action u.

4.2. Example 2: Power System Application. This section will provide a power system to test the proposed learning algorithm; thus, we choose $x = [\varsigma_1, \varsigma_2, \varsigma_3] \in \mathbb{R}^3$ as system states, where $x_1 = \varsigma_1$ is the incremental change of the frequency deviation, $x_2 = \varsigma_2$ defines the generator output, and $x_3 = \varsigma_3$ denotes the governor value position. Therefore, the state-space expression of this power system can be given as

$$\dot{x} = \begin{bmatrix} -\dfrac{1}{T_G} & 0 & -\dfrac{1}{F_r T_G} \\[2ex] \dfrac{K_t}{T_t} & -\dfrac{1}{T_t} & 0 \\[2ex] 0 & \dfrac{K_g}{T_g} & -\dfrac{1}{T_g} \end{bmatrix} x + \begin{bmatrix} \dfrac{1}{T_G} \\[2ex] 0 \\[1ex] 0 \end{bmatrix} u, \tag{32}$$

then we can give some parameters of the proposed power system as follows.

$T_G = 5(Hz/MW)$ is the time of the governor, $T_t = 10(s)$ denotes the time of the turbine model, $T_g = 10(Hz/MW)$ is the time of the generator model, $F_r = 0.5(s)$ indicates the feedback regulation constant, $K_t = 1(s)$ is the gain constant of the turbine model, and $K_g = 1(s)$ shows the gain constant of the generator model.

In order to complete this simulation, one assumes that this system is disturbed by an uncertain term as example 1. The initial system states are set as $x_0 = [-0.3, 0.5, 1]^T$, $Q = I$, and $R = 1$; the learning parameters are given as $\ell = 0.5$ and $\kappa = 100$. Similar to example 1, the offline solution of ARE can be given as

$$P^* = \begin{bmatrix} 2.5817 & 1.4963 & -1.7575 \\ 1.4963 & 7.7916 & 3.2394 \\ -1.7575 & 3.2394 & 11.4129 \end{bmatrix}. \tag{33}$$

Figure 6 shows the convergence of estimated matrix P; based on the offline solution given in (33), we have that the estimated solution P can converge to its optimal solution P^*; this in turn affects the system state response (as shown in

Figure 7). Figure 7 gives the system state response, which is smooth and bounded. The system control input is given in Figure 8.

5. Conclusion

In this paper, an online data-driven ADP method is proposed to solve the robust control problem for continuous-time systems with uncertainties. The robust control problem can be transformed into the optimal control problem. A new online ADP scheme is then introduced to obtain the solution of ARE via using the vectorization operator and Kronecker product. Finally, the closed-loop system stability and the convergence of the robust control solution are all analyzed. Simulation results are presented to validate the effectiveness of the proposed algorithm. It is worth noting that the research results are satisfied to the matched uncertainty condition. In our future work, we will extend the proposed idea to address the robust tracking control problem, which allows to carry out practical experimental validations based on existing test-rigs in our lab.

Data Availability

Data were curated by the authors and are available upon request.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

References

[1] F. Lin, R. D. Brandt, and J. Sun, "Robust control of nonlinear systems: compensating for uncertainty," *International Journal of Control*, vol. 56, no. 6, pp. 1453–1459, 1992.

[2] K. Zhou and J. Doyle, "Essentials of robust control," *Prentice Hall NJ*, vol. 38, 1998.

[3] J. Zhao, J. Na, G. Gao, and Y. Zhang, "Robust tracking control of uncertain nonlinear systems with adaptive dynamic programming," *Neurocomputing*, vol. 471, pp. 21–30, 2021.

[4] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, 2005.

[5] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, 2012.

[6] P. J. Werbos, "Advanced forecasting methods for global crisis warning and models of intelligence," *General Systems Yearbook*, vol. 22, no. 6, pp. 25–38, 1977.

[7] B. Xu, C. Yang, and Z. Shi, "Reinforcement learning output-feedback NN control using deterministic learning technique," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 25, no. 3, pp. 635–641, 2014.

[8] F. Wang, "Parallel control: a method for data-driven and computational control," *Acta Automatica Sinica*, vol. 39, no. 4, pp. 293–302, 2013.

[9] F. Lin, *Robust Control Design: An Optimal Control Approach*, John Wiley& Sons, 2007.

[10] D. Wang, D. Liu, C. Mu, and Y. Zhang, "Neural network learning and robust stabilization of nonlinear systems with dynamic uncertainties," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 4, pp. 1342–1351, 2018.

[11] J. Zhao, J. Na, and G. Gao, "Adaptive dynamic programming based robust control of nonlinear systems with unmatched uncertainties," *Neurocomputing*, vol. 395, pp. 56–65, 2020.

[12] P. Gahinet, A. Nemirovskii, A. J. Laub, and M. Chilali, "The LMI control toolbox," *In Proceedings of 1994 33rd IEEE Conference on Decision and Control*, vol. 3, pp. 2038–2041, 1994.

[13] J. Na, B. Wang, G. Li, S. Zhan, and W. He, "Nonlinear constrained optimal control of wave energy converters with adaptive dynamic programming," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 7904–7915, 2019.

[14] Y. Lv, J. Na, Q. Yang, X. Wu, and Y. Guo, "Online adaptive optimal control for continuous-time nonlinear systems with completely unknown dynamics," *International Journal of Control*, vol. 89, no. 1, pp. 99–112, 2016.

[15] H. Modares and F. L. Lewis, "Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning," *IEEE Transactions on Automatic Control*, vol. 59, no. 11, pp. 3051–3056, 2014.

[16] J. Na and H. Guido, "Online adaptive approximate optimal tracking control with simplified dual approximation structure for continuous-time unknown nonlinear systems," *IEEE/CAA Journal of Automatica Sinica*, vol. 1, no. 4, pp. 412–422, 2014.

Research Article

Reliability Importance Measures considering Performance and Costs of Mechanical Hydraulic System for Hydraulic Excavators

Wenting Liu [ID],[1] Qingliang Zeng [ID],[1,2,3] Lirong Wan,[1] Jinxia Liu,[1,2] and Hanzheng Dai [ID][1]

[1]*Department of Mechanical and Electronic Engineering, Shandong University of Science and Technology, Qingdao, Shandong 266590, China*
[2]*Shandong Province Key Laboratory of Mine Mechanical Engineering, Shandong University of Science and Technology, Qingdao 266590, China*
[3]*Department of Information Science and Engineering, Shandong Normal University, Jinan, Shandong 250358, China*

Correspondence should be addressed to Qingliang Zeng; qlzeng@sdust.edu.cn

Although some reliability importance measures and maintenance policies for mechanical products exist in literature, they are rarely investigated with reference to weakest component identification in the design stage and preventive maintenance interval during the life cycle. This paper is mainly study reliability importance measures considering performance and costs ($\mathrm{RIM_{PC}}$) of maintenance and downtime of the mechanical hydraulic system (MHS) for hydraulic excavators (HE) with energy regeneration and recovery system (ERRS) and suggests the scheduled maintenance interval for key components and the system itself based on the reliability $R_i(t)$. In the research, the required failure data for reliability analysis is collected from maintenance crews and users over three years of a certain type of hydraulic excavators. Minitab is used for probable distribution estimation of the mechanical hydraulic system failure times, and the model is verified to obey Weibull distribution. $\mathrm{RIM_{PC}}$ is calculated by multiplying the reliability $R_i(t)$ and weighting factor W_i and then compared with other classical importance measures. The purpose of this paper is to identify the weakest component for MHS in the design stage and to make appropriate maintenance strategies which help to maintain a high reliability level for MHS. The proposed method also provides the scientific maintenance suggestion for improving the MHS reliability of the HE reasonably, which is efficient, profitable, and organized.

1. Introduction

With development of society and the progress of science and technology, crisis of lack of energy and serious environmental pollution has become increasingly prominent. As the second-largest internal combustion engine product in addition to autoindustry, construction machinery pollutes environment more seriously than other industries, since its large engine capacity, high oil consumption, and high emissions. To achieve energy conservation, pollution reduction, and sustainable development, various energy-saving technologies have been applied in construct machinery, such as hybrid, energy recovery, electronic control, and new energies. Among these, the favorite for customers and manufacturers is energy recovery technology, for its low cost and high production efficiency. The hydraulic system of con-

struction machines become more complicated when upgraded with energy recovery unit; hence, quality and reliability analyses for complex hydraulic system become the most important task in the stages of design, running, and maintenance. Importance measures are utilized to evaluate the effect of parts on a system when single or multiple parts fail or their states change; they are functions of reliability parameters and system structures. In system design stage, the weakest part of system could be sought out by importance analysis, which applied for supporting system promotion from a design perspective. In system operational stage, the preventive maintenance policies or replacement scheme can be performed in right time by means of important measures analysis, which could ensure system operated normally.

In this study, importance measure calculation of individual component which belongs to the subsystem is used to

measure the effectiveness of the reliability for complex mechanical hydraulic system with energy recovery system. Reliability models are established for important measure calculation, and some assumptions are made as follows; a binary system is formed from two functional states: perfect functionality and complete failure. For energy regeneration and recovery system (ERRS) of construction machinery have the characteristics of multicircuit, nonlinearity, and uncertainty, it is difficult to do reliability analysis and reliability design optimization in practical production. The RIM_{PC} is proposed for importance measure, and prevention scientific maintenance is suggested for improving the ERRS reliability of the HE reasonably. It is significant to do the important analysis for key subsystems of complex system, for manufacturers to put their effort to the analyzed main parts.

The major contributions of this paper include the following:

(1) Assessment index RIM_{PC} is presented to evaluate system reliability; it is convenient and practical for maintenance crew

(2) Develop a new reliability importance measure and identify the manufacturing bottleneck of energy regeneration and recovery system assessment of construction machinery in the design stage

(3) Suggest appropriate preventive maintenance interval of system for maintenance crew to keep high reliability for new system

The upcoming sections will cover the following: Section 2 reviews prime importance measures briefly for the binary and multistate systems. In Section 3, a new reliability importance measure is proposed, and preventive maintenance interval is suggested. An energy regeneration and recovery system of hydraulic excavator is taken as illustration in Section 4 to explain how the proposed measure works and then discovers by new method, and various importance measures are compared and discussed. The conclusion comes in Section 5.

2. Review of Importance Measures for Binary and Multistate Systems

Numerous importance measures and reliability assessment methods have been developed in recent years, like Birnbaum method and the optimization measures, Monte Carlo simulation, Markov chain, and Fault Tree Analysis, most of which are utilized in the field of electronics and aerospace [1, 2].

This section reviews kinds of classic importance measures in reliability system design. Birnbaum proposed the classic binary importance measures of components in a coherent system in the 1960s [3], categorizing importance measures into three classes, namely, the structure importance measure, the reliability importance measure, and the lifetime importance measure. Recent advances and extensions to multistate components on importance measures

have been successfully developed and applied for various purposes as shown in the literature [4].

Lambert conducted on fault trees for decision-making in system analysis and criticality importance measure [5]; Vesely and Fussell implemented Fussell-Vesely importance measure [6, 7], concerned with component failures contributing to system failure, which refers to the probability of system failure when at least one of the minimum cut sets fails, and represented the ratio of the minimum cut set of component failure to system failure. Armstrong and Hong introduced joint reliability importance of components and k-out-of-n systems and analyzed the influence of primary and secondary components on system reliability [8, 9].

Binary decision diagram is a method proposed by Akers in the 1970s and developed in recent years based on fault tree analysis [10], for the advantages such as in low computational complexity and easy implementation, this methodology is popularly utilized in practical applications [11–17].

Barlow and Wu defined a system state function for coherent systems with multistate components and investigated its properties. They supposed that the results for the theory of binary structures could be applied in multistate component fault by natural extensions in terms of system function [18].

Lisnianski et al. defined multistate systems (MSSs) as they had different performance levels and several failure modes with various effects on the entire system's performance. He reviewed methods and tools used in the field of reliability assessment, optimization, and application [19]. The research team also did a lot of work in solving a family of MSS problems, such as structure optimization, optimal expansion, maintenance optimization, and optimal multistage modernization. And they also proposed an approach based on the universal generating function technique for the evaluation of some commonly used importance measures. [20, 21] presented a new method of dynamic availability and perform ability analysis for a large-scale multistate system based on robotic sensors [22, 23].

The composite importance measure proposed by Ramirez-Marquez and Coit about importance measures was to disclose critical part in a system so that the maintenance crew could rank the components in a system by means of their impact to performance reduction and production loss [24].

Natvig presented a probability model of operations and maintenance, described various types of MSSs, and searched on the measures of component importance in nonrepairable and repairable multistate strongly coherent systems [25, 26].Wu et al. proposed new utility importance of a component state in MSS, clarified the difference with importance measures suggested by William S. Griffith, and overcome some drawbacks. They also discussed the impact of an individual part to the performance utility of an MSS, so as to optimize it [27].

Zhang developed a heuristic policy for maintaining multistate systems for allocating maintenance resources to systems with higher importance [28]. The criticalities of different parts and the long-term effects of successful maintenance activities on the throughput of a production system
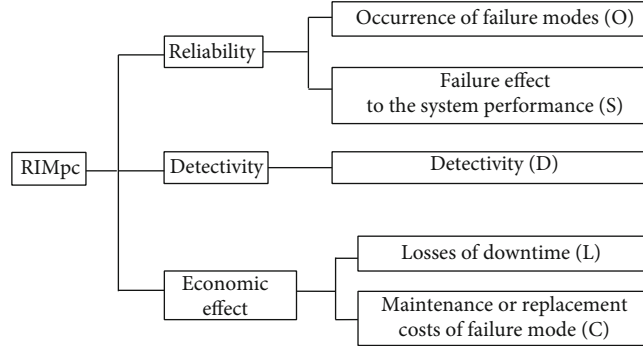
Figure 1: Reliability importance measure.

in a certain period to be solved by Ahmed and Liu and two types of importance measures prioritize the critical parts in the maintenance schedule to be presented [29]. Dao and Zuo presented some models based on reliability analysis to figure out the reliability of a complex system and assigned the reliabilities of its parts in a range of states varying from perfect functioning to complete failure [30].

Do and Bérenguer developed a novel time-dependent importance measure that could be utilized to rank the parts or groups of parts through their ability and to promote the system reliability for a given mission according to the conditional reliability evaluation of the system [31]. Borgonovo introduced the differential importance measure, a new sensitivity measure for probabilistic safety assessment [32, 33], proposed a new importance measure for time-independent reliability analysis, and offered a rank comparison with other time-dependent and time-independent reliability importance measures [34].

Peng et al. proposed two new importance measures for systems with S-independent degrading components and with S-correlated degrading components considering the continuously changing status of the degrading components and the correlation between components [35]. Ahmadi et al. evaluated the reliability, availability, and maintainability of the tunneling equipment and analyzed the material hauling system in an earth pressure balance tunnel boring machine [36]. Proper importance measures can help to identify design weakness or operation bottlenecks, conduct optimal modifications for system upgrades and maintenance, and provide information about the importance of components on the system performance, which includes reliability, availability, productivity, safety, and detectability [37].

3. Proposed Method

In this paper, a new reliability importance measure considering performance of mechanical hydraulic system (MHS) and cost of maintenance and downtime of construction machine caused by MHS' failures is proposed for the whole machine whose reliability and performance can be improved effectively if the weakest part is predicted as early as possible. For complex systems, limited resources are supposed to allocate according to how important the components are to the system in the design, enhancement, and maintenance stage

efficiently. In this study, an optimal strategy is implemented economically to identify the improvable part for system performance taking into system reliability, operation performance, maintenance cost, and losses in downtime account. Figure 1 is block diagram of the proposed reliability importance measure.

3.1. Explanation of the Principle of Reliability Importance Measures. All components and a system under consideration have the set of reliability states $s = \{0, 1, \cdots, K\}$ and $s \geq 1$. The state of component i is ordered as $s_i(k)$. $s_i(0)$ denotes component i as complete failure, $s_i(k)$ denotes component i as perfect function, and states degrade with time t. The probability matrix P of the components is presented as P; for all components, $k \in s$, $0 \leq p_{ik} \leq 1$, and in each row P adds up to 1.

$$P = [p_{ik}] = \begin{bmatrix} p_{00} & \cdots & p_{0K} \\ \vdots & \ddots & \vdots \\ p_{K0} & \cdots & p_{KK} \end{bmatrix},$$

$$\mathrm{RIM_{PC}} = R_i(t, \cdot). \tag{1}$$

$R_i(t, \cdot) = [R_i(t, 0), R_i(t, 1) \cdots, R_i(t, k)]$ is called the multistate reliability function of a component i, $R_i(t, \cdot) = P$, where $R_i(t, \cdot)$ is calculated based on the Weibull model of components' historical failure data. The Weibull distribution is used to transform the data effectively to Weibull model in machine reliability analysis, which shows effective ability of describing the wear-out failures and the product lifetime. The mathematical expressions of the Weibull distribution are shown in Appendix A. Reliability analysis based on Weibull approach probably be considered to generate better solution when system reliability expectation is high [38–40]. The weighting factor W_i in Equation (1) is used to calculate $\mathrm{RIM_{PC}}$ which takes performance and cost of machine operation into account in after-sales stage. Suppose the ith component has n kinds of failure modes, where W_i is defined as follows:

$$W_i = \sum_{j=1}^{n} S_{ij} O_{ij} D_{ij} L_{ij} C_{ij}. \tag{2}$$

TABLE 1: Suggested ranking system for the occurrence of failure modes [37].

Rank (O_{ij})	Comment	Failure frequency	Coefficient
10	Extremely high	≥0.2	0.1
9	Very high	≥0.1	0.2
8	Repeated failures	≥0.05	0.3
7	High	≥0.03	0.4
6	Moderately high	≥0.02	0.5
5	Moderate	≥0.01	0.6
4	Relatively low	≥0.005	0.7
3	Low	≥0.003	0.8
2	Remote	≥0.0005	0.9
1	Nearly impossible	≥0.0001	1.0

TABLE 2: The ranking of downtime losses caused by the failure mode.

Losses	Ranking	Coefficient
<0.5	1	1.0
0.5-1	2	0.9
1-1.5	3	0.8
1.5-2.0	4	0.7
2.0-2.5	5	0.6
2.5-3.0	6	0.5
3.0-3.5	7	0.4
3.5-4.0	8	0.3
4.0-4.5	9	0.2
4.5-5.0	10	0.1

S_{ij}: The severity of the failure modes effect to the system performance, 10 ranks

O_{ij}: Occurrence of the failure modes, 10 ranks

D_{ij}: Detectability of the failure modes, 10 ranks

L_{ij}: Losses of downtime under the failure mode, losses of downtime of the machine include economic losses of the systems stop running, 10 ranks

C_{ij}: Costs of the maintenance or replacement when the failure happens, 10 ranks.

The detailed ranking of S, O, D, L, and C is shown in Tables 1–5, which are worked out with the database belonging to HE manufacture. It can also be used for other construction machines after being revised.

3.2. New Approach for ERRS Preventive Maintenance Interval. It is necessary to make proper preventive maintenance strategies in the design stage to reduce machine downtime, increase operation time, and improve the availability of the equipment during use. There are three main types of maintenance in machine life cycle management. One is routine maintenance; it is easy to implement with less cost; the second one is restorative maintenance, which requires low cost and short time; and the third one is replacement maintenance, which replaces the parts that have lost their functions and makes the equipment repair as new. Hydraulic excavators are usually used in harsh environment with higher failure rates, so that the economic benefits for users are affected if as the traditional maintenance plan.

According to the calculated reliability of the old excavator hydraulic system, new maintenance method is put forward to guide the maintenance of energy recovery system, reduce the failure rate, improve the service life, and make users gain greater economic benefits. Moreover, the study results can help to improve excavators manufactures' maintenance management, to change users' one-sided understanding of excavator hydraulic system management, operation, maintenance, and other technical requirements, and further, to improve the reliability of the whole machine.

According to the standard regulation of the construction equipment maintenance, the driver performs routine maintenance per shift, and maintenance crew implement restor-

ative maintenance per 200 hours, replacement maintenance per 600 hours, and overhaul per 1800 hours. Most of the manufacturers recommended maintenance intervals at operation time of machines are 250 hours, 500 hours, 1000 hours, 2000 hours, 4000 hours, and 5000 hours, respectively. The predictive maintenance process proposed in this paper is shown in Figure 2.

Routine maintenance T_M is the same as traditional maintenance per shift, and preventive maintenance T_P is defined as per 500 hours. Restorative maintenance T_{F1}, T_{F2} $..., T_{Fn}$ is determined by the value of $R_i(t)$ at the operation time t. If the $R_i(t)$ of one component in the system is lower than the $R_{set}(t)$, which was described in the paper [38], the first restorative maintenance T_{F1} should be taken. Since there is time-delay for $R(t)$ rising, $R(t)$ will decrease first and then rise after restorative maintenance but not as high as initial value. T_F is decreasing with increasing usage time of the machine, so all the values of the T_{Fn} are different and gradually decrease. Replacement maintenance T_R is implemented at the time when the components' $R(t)$ achieves the minimum value as the preset. Overhaul period T_D is determined as

$$T_D = \left(\frac{T}{t_0} \cdot \frac{\gamma}{\beta - 1} \right)^{1/\beta}. \tag{3}$$

T: Denotes the average maintenance time

t_0: Denotes the average routine maintenance time

β: Denotes the estimated shape parameter of maintenance parts.

The scheduled maintenance time of ERRS is shown in Figure 3.

4. Unit Case Study

4.1. Description of Energy Regeneration and Recovery System for Hydraulic Excavator. The case studied in this paper MHS with energy regeneration and recovery system (ERRS) which is newly developed and used in hydraulic excavators. The ERRS is designed based on the balancing theory; the schematic principle of HE with ERRS is shown in Figure 4 [41,

TABLE 3: The ranking of components maintenance or replacement costs (10^3 RMB).

Costs	Ranking	Coefficient
<1.0	1	1.0
1.0~5.0	2	0.9
5.0~10.0	3	0.8
10.0~15.0	4	0.7
15.0~20.0	5	0.6
20.0~25.0	6	0.5
25.0~30.0	7	0.4
31.0~35.0	8	0.3
35.0~40.0	9	0.2
40.0~45.0	10	0.1

42]. Reversing valves 6, 11, and 12 are moving to the left side when the boom goes down, then the hydraulic oil(HO) is pumped into the rod cavity of main boom cylinder (RCMBC) 10 through reversing valve 6. One branch of HO is carried into the two rod cavity of balance cylinders (RCBC) 9 via reversing valve 12; another branch of HO in the piston cavity of main boom cylinder (PCMBC) 10 returns to the tank by reversing valve 11; and the self-gravity potential energy generated during the boom down is accumulated into hydraulic accumulator (HA) as hydraulic energy via valve 7 [39]. The hydraulic accumulator (HA) is used for storing and releasing energy; accumulator's pressure acting on the boom always shows itself as a balancing weight for the load [43].

Reversing valves 6, 11, and 12 are all linked on the right side when the boom goes up; then, HO is pumped from the tank into the PCMBC 10 by reversing valve 6, HO accumulated in accumulator is released into RCBC 9, and HO in RCMBC 10 and RCBC 9 return to the tank through reversing valve 11 and valve 12, respectively [42].

For a complex mechanical hydraulic system, the system reliability is based on the component reliability. It is critical to know the importance of each part of MHS; severe failure of the component may lead to collapse of the whole system if it had not been discovered in time. Various factors in the process of maintenance must be considered, such as maintenance cost, difficulty, and time [44, 45].

For example, leakage of hydraulic cylinder will reduce the work efficiency of MHS; before any obvious fault occurs, it must be anticipated with preventive measures. Any one tiny failure of subsystem may cause the failure of the entire system if there are no backups for these components.

4.2. Schematic Diagram of the Mechanical Hydraulic System. A schematic diagram of the MHS of ERRS is illustrated in Figure 5. Some components of the system are unlikely to fail during the machine lifetime, as known from engineering experience, like throttle valves and solenoid valves. Therefore, these kinds of components are not conducted importance analysis in this work. However, servo valves, cylinders, pumps, reversing valves, booms, tubes, and accumulators, which with higher failure rates throughout the whole energy recovery and release process, are most likely to be vulnerable components.

In actual operation of excavators with ERRS, all the hydraulic components do not have backups due to high cost. How to balance the system reliability improvement and cost reduction is very important for excavator manufactures.

4.3. Calculation of Reliability Importance Measures. To study the importance and identify the weakest components of the MHS, this paper collects the failure data of the 30 Ton HE for three years from the maintenance database. The number of working HE in all is 973, recorded by GPS, and the number of failure data of the mechanical hydraulic system is 197.

In this case, the following assumptions are made for mechanical hydraulic components and system:

(I) All components have two states: functioning and failed

(II) All components are in a perfect state at the beginning

(III) Behaviors of components are mutually statistically independent

(IV) All the components are repairable to new when they fail

As a universally adaptive distribution, the Weibull law is widely used to describe the life distribution of mechanical products for modeling the failure behavior of components. Minitab is used to fit all the failure times of MHS and to test the Anderson-Darling goodness. Anderson-Darling (AD) test is a kind of square-variance statistics. Although the statistical process is slightly complicated, it can maintain good performance when using the small sample size. The fitted results are shown in Table 6 and Figure 6. As shown in Table 7, the three-parameters Weibull distribution has the smallest AD statistics, with the value of 0.493, so it has been clearly seen that the best goodness of fit is the three-parameter Weibull distribution for ERRS, and the components of this ERRS testified to be fitted as a three-parameter Weibull distribution well.

This paper uses the mean rank order methods to calculate the empirical cumulative distribution function of each component of MHS and the reliability at 3000 hours, because the warranty services of repair are during 3000 hours for machine manufactures.

$$F_m(t) = \frac{j(f_m) - 0.3}{n + 0.4}, \tag{4}$$

$$j(f_m) = j(f_{m-1}) + n_{fm} \cdot N(f_m), \tag{5}$$

$$N(f_m) = \frac{(n+1) - j(f_{m-1})}{1 + (n - n_{fm} - n_{su})}. \tag{6}$$

$j(f_m)(m = 1, 2, \cdots n_f)$
n:Sample number
n_f:Failure number
n_s:Unfailed number.

TABLE 4: Suggested ranking system for the severity of failure modes [38].

Failure effect	Failure criterion	Ranking	Coefficient
Inconsistent with the safety legislation or the regulations	Hazardous without warning potential safety, health, or environmental issue	10	0.1
	Failure will occur with warning potential safety, health, or environmental issue	9	0.2
Disruption or decline to facility function	The machine runs malfunctioning	8	0.3
	The machine runs properly but moderate disruption to facility function	7	0.4
Disruption or decline to secondary function	Some portion of secondary function is lost	6	0.5
	Moderate disruption to secondary function	5	0.6
Appearance or noise and other functions is poor	Some portion of process is delayed	4	0.7
	Most users (>75%) likely to complain	3	0.8
No discernible effect on safety, environment, or mission	More than half (>50%)of the users likely to complain	2	0.9
	a few users (>25%) likely to complain	1	1.0

TABLE 5: Detectivity evaluation criteria.

Detectivity	Control	Ranking	Coefficient
Absolute uncertainty	No control. Design control will not and/or cannot detect a potential cause/mechanism and subsequent failure mode	10	1.0
Very remote	Very remote chance the design controls will detect a potential cause/mechanism and subsequent failure mode	9	0.9
Remote	Remote chance the design controls will detect a potential cause/mechanism and subsequent failure mode	8	0.8
Very low	Very low the design controls will detect a potential cause/mechanism and subsequent failure mode	7	0.7
Low	Low chance the design controls will detect a potential cause/mechanism and subsequent failure mode	6	0.6
Moderate	Moderate chance the design controls will detect a potential cause/mechanism and subsequent failure mode	5	0.5
Moderately high	Moderately high chance the design controls will detect a potential cause/mechanism and subsequent failure mode	4	0.4
High	High chance the design controls will detect a potential cause/mechanism and subsequent failure mode	3	0.3
Very high	Very high chance the design controls will detect a potential cause/mechanism and subsequent failure mode	2	0.2
Almost certain	Design controls will almost certainly detect a potential cause/mechanism and subsequent failure mode	1	0.1

The parameters β, γ, and η of cylinders, pump, boom, reversing valve, tubes, and hydraulic system are fitted by Origin. Figure 7(b) shows cylinders' $R_i(t)$ changes with operation time, $R_{ci}(t) = 0.9311$ when after 3000 hours operation time. Figure 7(c) shows pumps' $R_{pi}(t)$ changes with operation time, $R_{pi}(t) = 0.9798$ when after 3000 hours operation time. Figure 7(d) shows the reversing valves' $R_{Ri}(t)$ changes with operation time, $R_{Ri}(t) = 0.9826$ when after1500 hours operation time, for no more failure data about reversing valves from 1500 to 3000 hours, so here, $R_{Ri}(t) = 0.9826$ is reckoned stayed there ever until 3000. Figure 7(e) shows booms' $R_{Bi}(t)$ changes with operation time, $R_{Bi}(t) = 0.9301$ when after 3000 hours operation time. Figure 7(f) shows the high-pressure tubes' $R_{Ti}(t)$ changes with operation time,

$R_{Ti}(t) = 0.9385$ when after 2000 hours operation time, for no more failure data about reversing valves from 2000 to 3000 hours, so here, $R_{Ri}(t) = 0.9385$ is reckoned stayed there ever until 3000. Figure 7(a) shows the hydraulic systems' $R(t)$ changes with time, $R(t) = 0.7636$ when after 3000 hours operation time. And parameters β, γ, and η values are known from Figure 7. Then, $R_i(t)$ is calculated by the parameters, the $R_i(t)$ of the accumulator and servo valve is calculated by Equation (4) for its few failure numbers (as is shown in Table 8).

The importance of components (I_{PC}) is obtained by (1) and (2); they are listed in Table 9. s_{ik}, o_{ik}, d_{ik}, l_{ik}, and c_{ik} are designated based on Tables 1–5. The cylinder, pump, and boom have two kinds of failure modes; the
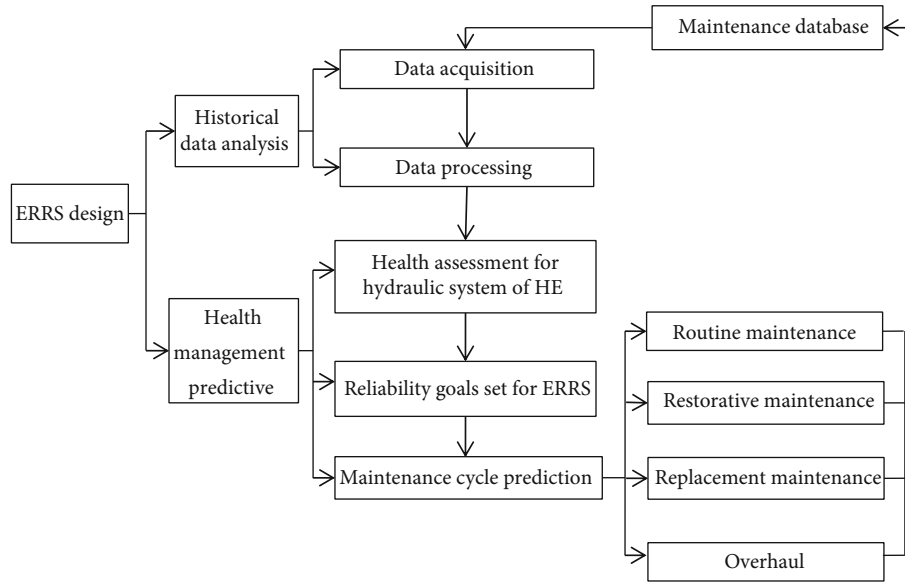
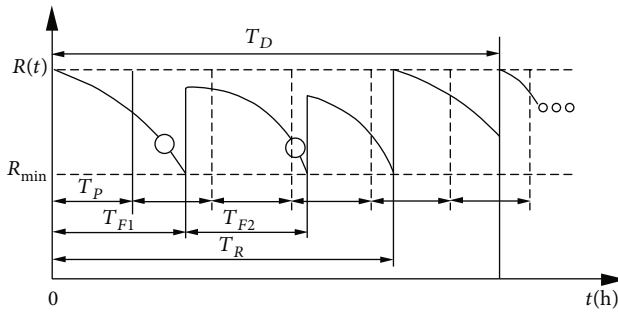FIGURE 2: Predictive maintenance process of ERRS.

FIGURE 3: Scheduled maintenance time of ERRS.

accumulator, servo valve, reversing valve, and tube only have one failure mode.

As shown in Table 10, the component boom and accumulator have the largest and smallest importance ranking order, respectively, in all different importance measures. This means the boom is the least reliable unit, and the accumulator is the most reliable unit in the MHS. The other component ranking orders change as importance measures change, but the components importance ranking order is completely the same in the method Time Integral Importance Measures (TIIM) and the Criticality Reliability Importance of Component for system failure.

I_{TIIM} is used to estimate components' importance better in their lifecycle and seek out the most responsible component for subsystem performance loss while ignoring the effects from the costs of the maintenance and downtime by the component failure modes. The criticality time-dependent lifetime importance for system failure at time t (I_{cf}) is defined as the probability when a component failure causes the given system failure; it does consider the performance losses and costs in the process of systems or products operating. The traditional Birnbaum importance measures do not consider the criticality and the variety of mean lifetime of a system caused by components.

The proposed method in this study considers the severity of the components failure, occurrence rate of the different components failure mode, difficulty level to detect the failure modes, maintenance costs, and breakdown losses when the components failure modes occur. All these aspects are expected to be taken into account in the new system design stage based on the predecessor. The $\mathrm{RIM_{PC}}$ can evaluate the importance of complex mechanical hydraulic system components more simply and effective by historical database compared with other methods. From the definition of $\mathrm{RIM_{PC}}$, it can be used to conduct the importance evaluations not only in multistate system but also in binary system. Therefore, the conclusions derived from binary-state systems can also be used for multistate ones.

4.4. Suggestion about the Optimization of ERRS Design and Maintenance. According to the $\mathrm{RIM_{PC}}$ value of components of MHS shown in Table 9 and the ranking order shown in Table 10, accumulator and servo valve have higher reliability but lower failure rate. So, they are lower importance components in MHS boom, and cylinders have lower reliability but higher failure rate; they are higher importance components in MHS. The ranking order of pump, reversing valve, and tube is 3, 4, and 5, respectively.

Through the analysis of historical failure database, the main failure modes of the boom are fracture on the root and welds cracking between the side plates, because fatigue strength is insufficient and badly soldered. In MHS, a new structure of boom has been developed since balance cylinders increase, so methods of robust design optimization in the design stage and enhancement of welding quality in the manufacturing stage should be taken to the boom reliability promotion.

The main failure modes of cylinder include crack, leakage, abrasion, and creep. Dominant reasons for the failures are encounter external impact, instantaneous high pressure, hydraulic oil pollution, and unreasonably kinematic pair
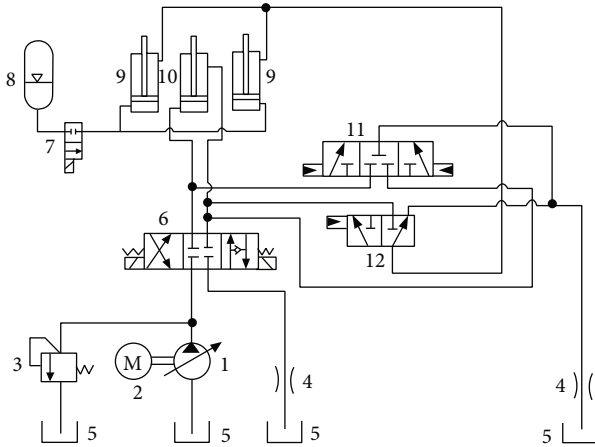
FIGURE 4: The schematic principle of ERRS. 1: pump; 2: engine; 3: relief valve; 4: throttle valve; 5: oil tank; 6, 7, 11, 12: reversing valves; : hydraulic accumulator; 9; balance cylinders; 10: boom cylinder.
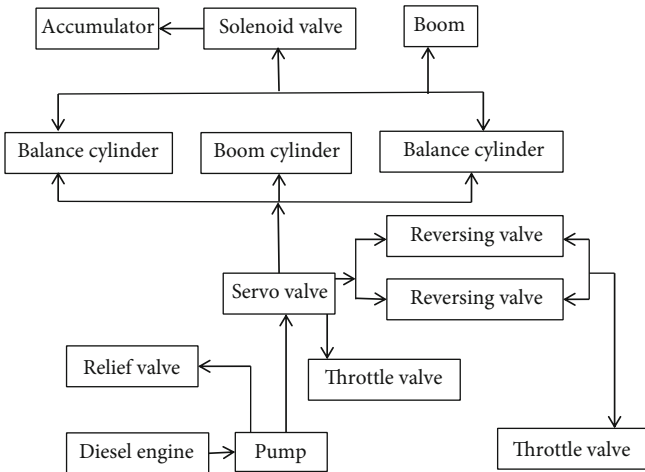

FIGURE 5: Block diagram of the hydraulic system.

clearances, respectively. A protective board suggested to be added on the top of HC which suffers intense impact easily and reduces the instantaneous high pressure caused by energy released from accumulator to the system in the design stage. Promoting assembling accuracy and strengthening the final inspection on the assembly line are also good choice for reliability improvement.

The main failure modes of the tubes are leakage and burst. The main reason of the failures is that overloaded transient impacts pressure in high-pressure tubes, which should be improved in MHS. The abnormal vibration of the hydraulic piston pump causes the leakage, noise, and cracking of the pump body; most of them occurred after 2000 hours of operation time. And when the occurrence is lower at the ranking 0.7, we suggest changing the maintenance interval to enhance pump reliability. The occurrence of reversing valve leakage can be reduced by optimizing seal quality; the failures of the accumulator and servo valve have occurred accidentally, with little effect on the reliability of the HEs. Further tracing will be performed.

TABLE 6: The number of failure for each component.

Component	Failure number
Cylinders	44
Accumulator	1
Servo valve	3
Pump	21
Boom	81
Reversing valve	12
Tubes	35
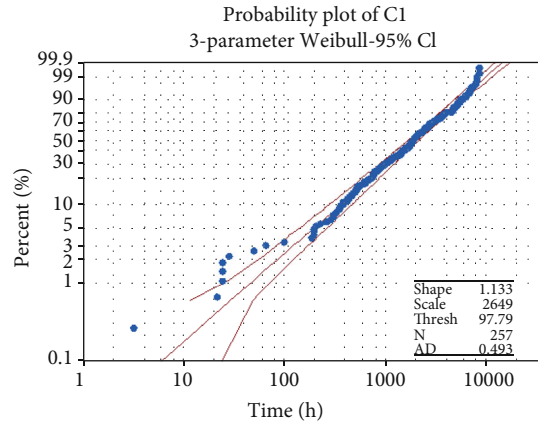
FIGURE 6: Probability plot of the hydraulic system of HE.

TABLE 7: Failure time distribution AD test for HS of HE.

Distribution	AD
Weibull	0.720
3-parameter Weibull	0.493
Normal	3.152
Lognormal	1.817
3-parameter lognormal	0.991
Gamma	0.598
Exponential	3.026
2-parameter exponential	1.554
Small extreme value	15.805
Large extreme value	3.854
Logistic	7.529
Loglogistic	1.525
3-parameter Loglogistic	1.374

4.5. Explanation of Scheduled Maintenance Time for Key Components of ERRS and System Itself. In this section, the scheduled maintenance time of boom will be shown, since it ranks the first in the importance list of MHS.

Routine maintenance T_M is set as the same as traditional maintenance time 8 hours per shift, and preventive maintenance T_P is chosen as per 500 hours.

Restorative maintenance T_{F1} of boom is the time when the value of R_{set} decreases to 0.9, so T_{F1} is determined with parameters' estimated value of β, γ, and η. It is
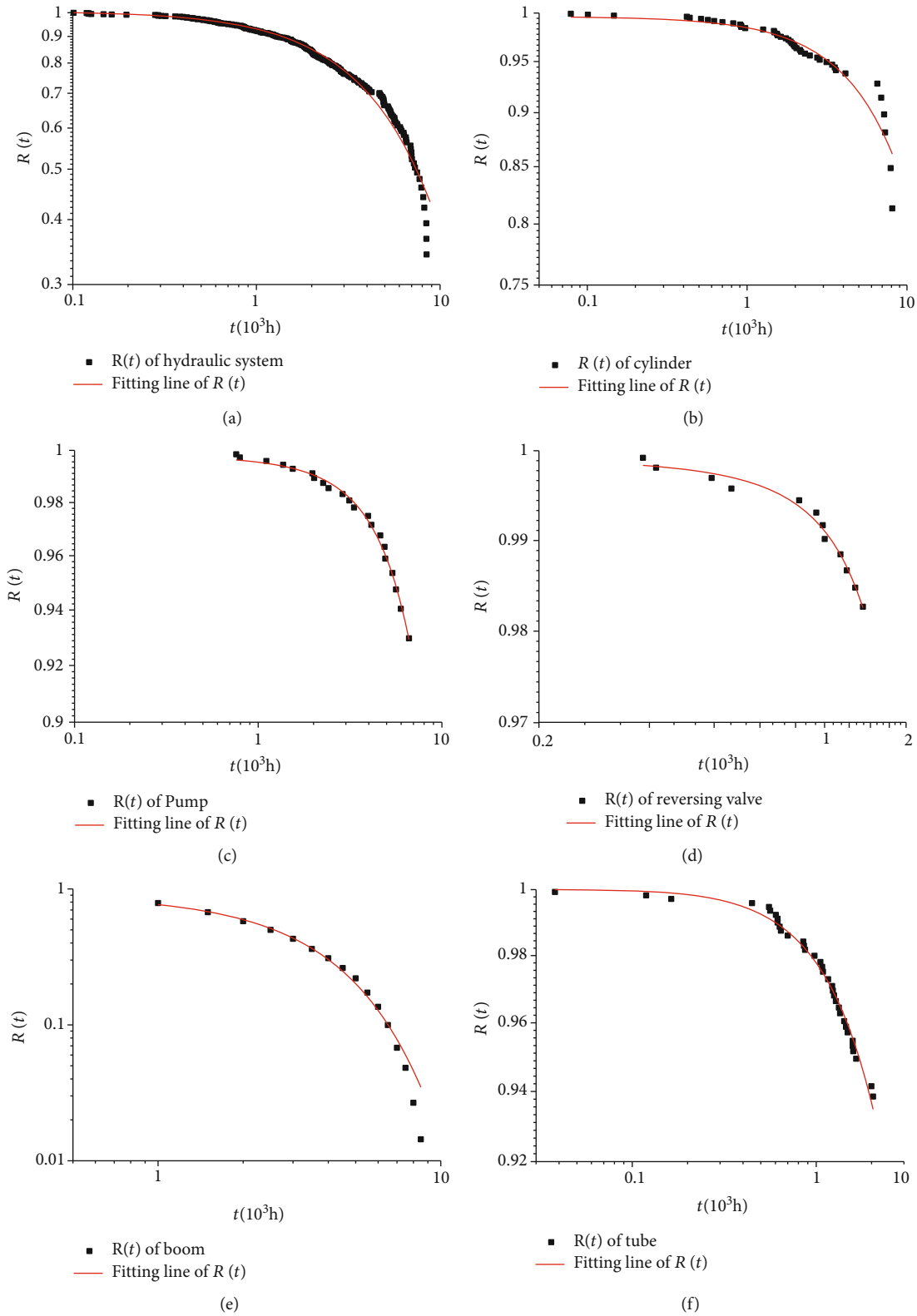
FIGURE 7: Reliability of the hydraulic system and key components.

approximately 3900 working hours. R_{min} of boom is set as increasing 20% as we have 0.5960 in Table 8, so the value R_{min} of boom is 0.7152, and replacement maintenance time T_R is approximately 7000 working hours. Overhaul period T_D of ERRS is determined as Equation (3). Parameters $\gamma = 101$ and $\beta = 1.08$ are obtained from Figure 7(a),

TABLE 8: Components reliability at $t = 3000$ hours.

| No.	Component (i)	β	γ	η	$F_i(t)$	$R_i(t)$
1	Cylinders	1.2689	-438.33	38267.0	0.0458	0.9542
2	Accumulator	/	/	/	0.0010	0.9990
3	Servo valve	/	/	/	0.0021	0.9979
4	Pump	2.8252	-2374.7	22680	0.0170	0.9830
5	Boom	1.5815	-884.82	4362.3	0.4040	0.5960
6	Reversing valve	1.6789	383.92	13075.2	0.0649	0.9351
7	Tubes	1.4985	37.88	12108.0	0.1143	0.8857

TABLE 9: Components importance considering performance and costs at $t = 3000$ hours.

Components	$R_i(t)$	s_{ik}	o_{ik}	d_{ik}	l_{ik}	c_{ik}	RIM_{PC}
Cylinder	0.9542	0.3	0.5	0.7	0.7	0.8	0.2364
		0.5	0.6	0.7	0.9	1.0	
Accumulator	0.9990	0.9	0.9	0.7	0.7	0.9	0.3569
Servo valve	0.9979	0.7	0.8	0.6	0.8	1.0	0.2682
Pump	0.9830	0.3	0.6	0.9	0.4	0.4	0.2483
		0.4	0.7	0.9	0.9	1.0	
Boom	0.5960	0.2	0.4	0.9	0.6	0.2	0.0963
		0.7	0.3	0.9	0.9	0.9	
Reversing valve	0.9351	0.7	0.6	0.9	0.8	0.9	0.2545
Tube	0.8857	0.4	0.4	0.9	1.0	1.0	0.2551

TABLE 10: Components Importance for different methods at $t = 3000$ hours.

Component	IB	Ranking	ITIIM	Ranking	Icf	Ranking	RIMPC	Ranking
Cylinder	0.0643	5	169.44	3	0.0131	3	0.2364	2
Accumulator	0.0007	7	193433.7	7	0.000003	7	0.3569	7
Servo valve	0.0339	2	4254.42	6	0.0009	6	0.2682	6
Pump	0.0464	4	1624.55	5	0.0038	4	0.2483	3
Boom	0.0973	1	30.299	1	0.0283	1	0.0963	1
Reversing valve	0.0462	3	990.35	4	0.0034	5	0.2545	4
Tube	0.0777	6	50.698	2	0.0203	2	0.2551	5

so $T_D = 7760$ working hours, where T is the average maintenance time 100 working hours, and t_0 is the average routine maintenance time 8 working hours.

5. Conclusion

This paper mainly discusses the RIM_{PC} (reliability importance measures based on performance and costs of maintenance and downtime). Firstly, the definition of RIM_{PC} of MHS' components is presented. Secondly, the proposed method is verified by the type of MHS with ERRS which belongs to 30 Ton HE. Thirdly, several classical importance measures are compared with the proposed method, and pros and cons are analyzed.

The major conclusions are summarized as follows:

(1) Although a components' deterioration from function to failure will go through many states, only functioning and failure are considered in the process of machine using, therefore, the multistate system has been simplified to a binary system for reliability importance analysis. RIM_{PC} can be used to estimate the component importance better in its lifecycle and seek out the most important component for system reliability. Then, more attention can be paid to the most important one to improve system performance and reliability efficiently

(2) RIM_{PC} can be used to estimate the importance of complex MHS' components of existing product and predict the reliability of the new-generation product based on existing product's historical failure data. It is also feasible to guide the designers to obtain some clues of reliability allocation in the design stage, to identify what is the root cause for the failure of the part at different operation stage, and to improve the robustness performance of the part in time

(3) And to guide the maintenance crew in assigning maintenance resources to achieve higher performance in a relatively long term for new systems and new products

(4) Determination of preventive maintenance interval for key components of MHS and system itself based on the historical reliability of them in the design stage can help maintenance crew to keep HE with ERRS functioning effectively

(5) Since the proposed importance measures are developed to evaluate components in a fixed construction machine HE, more research work should be done to find the effects to structural changes, product performance, and reliability improvement in future studies

Appendix

A. A Detailed Description Is Given

If the failure behavior of component I described by Weibull stochastic process, its lifetime follows a probability density function is shown as

$$ f(t) = \frac{\beta}{\eta} \left(\frac{t - \gamma}{\eta} \right) \exp^{-((t-\gamma)/\eta)^\beta}. \tag{A.1} $$

The probability distribution function is given by:

$$ F(t) = 1 - \exp \left[-\left(\frac{t - \gamma}{\eta} \right)^\beta \right]. \tag{A.2} $$

The reliability of component can be then evaluated by:

$$ R(t) = \exp \left[-\left(\frac{t - \gamma}{\eta} \right)^\beta \right], \tag{A.3} $$

where η, β, and γ denote the scale, shape, positional parameters of the components.

List of Symbols

HA: Hydraulic accumulator
HC: Hydraulic cylinders
ERRS: Energy regeneration and recovery system
MSSs: Multistate systems
FMEA: Failure mode and effect analysis
PCMBC: Piston cavity of main boom cylinder
$F_m(t)$:   Cumulative failure distribution function
W_i: Weighting factor of component i
S_{ij}:   The serious of the failure modes effect to the system performance failure modes j for component i
O_{ij}:   Occurrence of the failure modes j for component i
D_{ij}:   Detectability of the failure mode j for component i
L_{ij}:   Losses of downtime under the failure mode j for component i
C_{ij}:   Costs of the maintenance or replacement when the failure happens
HE: traditional excavators
HO: Hydraulic oil
$R_i(t)$:   Reliability of component i at time t
$s_i(t)$:   Reliability states of component i
RCMBC: Rod cavity of main boom cylinder
RCBC: Rods cavity of balance cylinders
β: Shape parameter of Weibull distribution
η: Scale parameter of Weibull distribution
γ: Location parameter of Weibull distribution
$j(f_m)$:   Average number of the failure time
n: Sample number
n_f: Failure number
n_s: Unfailed number
T_M: Routine maintenance
T_P: Restorative maintenance
T_R: Replacement maintenance
T_D: Overhaul period
T: Average maintenance time
t_0:   Average routine maintenance time.

Data Availability

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

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

References

[1] S. M. Wu and L. Y. Chan, "Performance utility-analysis of multi-state systems," *IEEE Transactions on Reliability*, vol. 52, no. 1, pp. 14–21, 2003.

[2] A. Lisnianski, "Extended block diagram method for a multi-state system reliability assessment," *Reliability Engineering and System Safety*, vol. 92, no. 12, pp. 1601–1607, 2007.

[3] Z. W. Birnbaum, *On the Importance of Different Components in a Multicomponent System in Multivariate Analysis*, P. R. Krishnaiah, Ed., Academic Press, New York, 1969.

[4] W. Kuo and X. Zhu, *Importance Measures in Reliability, Risk, and Optimization: Principles and Applications*, John Wiley and Sons, 2012.

[5] H. E. Lambert, *Fault Trees for Decision Making in Systems Analysis*, Univ, Calif., Livermore, Berkeley, UC, 1975, Ph. D. dissertation.

[6] W. E. Vesely, "A time-dependent methodology for fault tree evaluation," *Nuclear Engineering and Design*, vol. 13, no. 2, pp. 337–360, 1970.

[7] J. B. Fussell, "How to hand-calculate system reliability and safety characteristics," *IEEE Transactions on Reliability*, vol. -R-24, no. 3, pp. 169–174, 1975.

[8] M. J. Armstrong, "Joint reliability-importance of components," *IEEE Transactions on Reliability*, vol. 44, no. 3, pp. 408–412, 1995.

[9] J. S. Hong, H. Y. Koo, and C. H. Lie, "Joint reliability importance of _k_ -out-of- _n_ systems," *European Journal of Operational Research*, vol. 142, no. 3, pp. 539–547, 2002.

[10] S. B. Akers, "Binary decision diagrams," *IEEE Transactions on Computers*, vol. C-27, no. 6, pp. 509–516, 1978.

[11] P. N. Bibilo, "Decomposition of a system of incompletely specified Boolean functions defined with a binary decision diagram," *Automation and Remote Control*, vol. 75, no. 7, pp. 1173–1194, 2014.

[12] Y. Mo, "New insights into the BDD-based reliability analysis of phased-mission systems," *IEEE Transactions on Reliability*, vol. 58, no. 4, pp. 667–678, 2009.

[13] L. M. Bartlett and J. D. Andrews, "An ordering heuristic to develop the binary decision diagram based on structural importance," *Reliability Engineering and System Safety*, vol. 72, no. 1, pp. 31–38, 2001.

[14] R. Remenyte-Prescott and J. D. Andrews, "An enhanced component connection method for conversion of fault trees to binary decision diagrams," *Reliability Engineering and System Safety*, vol. 93, no. 10, pp. 1543–1550, 2008.

[15] C. Ibáñez-Llano, A. Rauzy, E. Meléndez, and F. Nieto, "A reduction approach to improve the quantification of linked fault trees through binary decision diagrams," *Reliability Engineering and System Safety*, vol. 95, no. 12, pp. 1314–1323, 2010.

[16] L. Xing and G. Levitin, "BDD-based reliability evaluation of phased-mission systems with internal/external common-cause failures," *Reliability Engineering and System Safety*, vol. 112, no. 4, pp. 145–153, 2013.

[17] X. Zang, N. Sun, and K. S. Trivedi, "A BDD-based algorithm for reliability analysis of phased-mission systems," *IEEE Transactions on Reliability*, vol. 48, no. 1, pp. 50–60, 1999.

[18] R. E. Barlow and A. S. Wu, "Coherent systems with multi-state components," *Mathmatics of Operations Reserch*, vol. 3, no. 4, pp. 275–281, 1978.

[19] A. Lisnianski, I. Frenkel, and Y. Ding, *Multi-State System Reliability and Optimization for Engineers and Industrial Managers*, Springer, London, UK, 2010.

[20] G. Levitin and A. Lisnianski, "A new approach to solving problems of multi-state system reliability optimization," *Quality and Reliability Engineering International*, vol. 17, no. 2, pp. 93–104, 2001.

[21] G. Levitin, L. Podofillini, and E. Zio, "Generalised importance measures for multi-state elements based on performance level restrictions," *Reliability Engineering and System Safety*, vol. 82, no. 3, pp. 287–298, 2003.

[22] A. Lisnianski, E. Levit, and L. Teper, "Short-term availability and performability analysis for a large-scale multi-state system based on robotic sensors," *Reliability Engineering and System Safety*, vol. 205, no. 1, pp. 107206–107209, 2021.

[23] A. Lisnianski, *Application of Extended Universal Generating Function Technique to Dynamic Reliability Analysis of a Multi-State System*, 2nd SMRLO, Shamoon Coll Engn, Beer Sheva, Israel, 2016.

[24] J. E. Ramirez-Marquez and D. W. Coit, "Composite importance measures for multi-state systems with multi-state components," *IEEE Transactions on Reliability*, vol. 54, no. 3, pp. 517–529, 2005.

[25] B. Natvig and I. Ebrary, *Multistate Systems Reliability Theory with Applications*, John Wiley and Sons, Ltd, Chichester, UK, 2011.

[26] B. Natvig, "Measures of component importance in non-repairable and repairable multistate strongly coherent systems," *Methodology and Computing in Applied Probability*, vol. 13, no. 3, pp. 523–547, 2011.

[27] J. Wu, T. S. Ng, M. Xie, and H. Z. Huang, "Analysis of maintenance policies for finite life-cycle multi-state systems," *Computers and Industrial Engineering*, vol. 59, no. 4, pp. 638–646, 2010.

[28] M. Zhang, "A heuristic policy for maintaining multiple multi-state systems," *Reliability Engineering and System Safety*, vol. 203, no. 1, p. 107081, 2020.

[29] A. A. A. Ahmed and Y. Liu, "Throughput-based importance measures of multistate production systems," *International Journal of Production Research*, vol. 57, no. 2, pp. 397–410, 2019.

[30] C. D. Dao and M. J. Zuo, "Selective maintenance for multistate series systems with S-dependent components," *IEEE Transactions on Reliability*, vol. 65, no. 2, pp. 525–539, 2016.

[31] P. Do and C. Bérenguer, "Conditional reliability-based importance measures," *Reliability Engineering and System Safety*, vol. 193, no. 1, p. 106633, 2020.

[32] E. Borgonovo and G. E. Apostolakis, "A new importance measure for risk-informed decision making," *Reliability Engineering and System Safety*, vol. 72, no. 2, pp. 193–212, 2001.

[33] E. Borgonovo, "Differential importance and comparative statics: an application to inventory management," *International Journal of Production Economics*, vol. 111, no. 1, pp. 170–179, 2008.

[34] E. Borgonovo, H. Aliee, M. Glaß, and J. Teich, "A new time-independent reliability importance measure," *European Journal of Operational Research*, vol. 254, no. 2, pp. 427–442, 2016.

[35] H. Peng, D. W. Coit, and Q. M. Feng, "Component reliability criticality or importance measures for systems with degrading components," *IEEE Transactions on Reliability*, vol. 61, no. 1, pp. 4–12, 2012.

[36] S. Ahmadi, S. Moosazadeh, M. Hajihassani, H. Moomivand, and M. M. Rajaei, "Reliability, availability and maintainability analysis of the conveyor system in mechanized tunneling," *Measurement*, vol. 145, no. 1, pp. 756–764, 2019.

[37] T. Niu, F. Li, B. Hu et al., "Research on the inverse problem of reliability evaluation–model and algorithm," *IEEE Access*, vol. 9, no. 1, pp. 12648–12656, 2021.

[38] A. Mahmoud, "Economic allocation of reliability growth testing using Weibull distributions," *Reliability Engineering and System Safety*, vol. 152, no. 1, pp. 273–280, 2016.

[39] G. S. Mudholkar, K. O. Asubonteng, and A. D. Hutson, "Transformation of the bathtub failure rate data in reliability for using Weibull-model analysis," *Statistical Methodology*, vol. 6, no. 6, pp. 622–633, 2009.

[40] S. Y. Grodzensky, "Reliability models based on modified Weibull distributions," *Measurement Techniques*, vol. 56, no. 7, pp. 768–774, 2013.

[41] W. T. Liu, Q. L. Zeng, L. R. Wan, and C. Wang, "A comprehensive method of apportioning reliability goals for new product of hydraulic excavator," *Mathematical Problems in Engineering*, vol. 2019, 11 pages, 2019.

[42] Q. Zeng, W. Liu, L. Wan, C. Wang, and K. Gao, "Maintenance strategy based on reliability analysis and FMEA: a case study for hydraulic cylinders of traditional excavators with ERRS," *Mathematical Problems in Engineering*, vol. 1, 11 pages, 2020.

[43] J. X. Liu, Z. Y. Jiao, F. X. Xian, and W. T. Liu, "Energy recovery and utilization system of excavator boom based on flow regeneration and balance theory," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 42, no. 1, pp. 1–11, 2020.

[44] J. Zhao, J. Na, and G. B. Gao, "Adaptive dynamic programming based robust control of nonlinear systems with unmatched uncertainties," *Neurocomputing*, vol. 395, pp. 56–65, 2020.

[45] J. Zhao, Y. F. Lv, and Z. L. Zhao, "Adaptive learning based output-feedback optimal control of CT two-player zero-sum games," *IEEE Transactions on Circuits and Systems II: Express Briefs*, p. 1, 2021.

Research Article

Robust Control of EHCS of Intelligent Commercial Vehicle under Load Change

Yushan Li [ID], Jitai Yu [ID], Ziliang Zhao [ID], and Bin Guo [ID]

Shandong University of Science and Technology, Qingdao, Shandong, China

Correspondence should be addressed to Ziliang Zhao; zhaoziliang1@sdust.edu.cn

Electro Hydraulic Coupling Steering (EHCS) system is a new type of intelligent commercial vehicle steering system, having strong nonlinear characteristics. Besides, the change of load would cause the change of control system parameters, making not easy to establish an accurate control model of it. To realize the robustness of EHCS under the change of load, the controller based on the adaptive control method is proposed in this paper. To this end, the cosimulation model of EHCS is first established, where the constructed control model is simplified to a 2-degree-of-freedom model under reasonable simplification and assumption. Then, the steering angle controller is designed based on the model reference adaptive theory. Finally, some simulations are given to show the effectiveness of the proposed control method.

1. Introduction

As commercial vehicles pay more attention to use efficiency and cost savings, intelligenceof it is more urgent than that of passenger vehicles. In addition, there is large volume, high load, and higher center of mass in commercial vehicles and the demand for driving safety is stronger [1]. The steering system is an important part of intelligent commercial vehicle, which directly affects the handling, stability, and driving safety. The steering system of commercial vehicles has gradually developed from mechanical to intelligent and electric power [2], which has gone through several stages: hydraulic power (HPS), electronically controlled hydraulic power (ECHPS), electric power (EPS), and steer-by-wire (SBW), e.g. The traditional HPS requires a lot of energy and has the disadvantage that the assist characteristics can not be adjusted [3], which will make the driver feel heavy when turning at low speed and float at high speed. ECHPS realizes the adjustable assist characteristics on the basis of HPS. EPS can only be installed on light trucks although it has a good road feel [4]. SBW has no real mechanical feedback, and safety redundancy measures are not easy to ensure safety for commercial vehicles. The electrohydraulic coupled steering (EHCS) system in this paper adds a set of power motor

unit on the basis of traditional HPS, which combines the advantages of high energy density of hydraulic system and high intelligence of electronic control system [5]. It can not only have good steering characteristics but also can be used as the hardware basis of active lateral control, matching the intelligent commercial vehicle.

EHCS is a complex nonlinear system, which has strong nonlinear characteristics such as hydraulic assistance and friction, e.g. [6]. Then, the load of the steering system will change with the vehicle speed, axle load, and other factors, and the parameters of the hydraulic system are time-varying, so it is not easy to establish an accurate control model [7–9]. Therefore, in order to control the angle of EHCS during autopilot, a simplified control model of EHCS must be established. Domestic and foreign scholars have done a lot of research on the steering system model. A 2-degree-of-freedom nonlinear hydraulic steering system model is proposed by Dell'Amico [10]. The system parameters are identified by the bench, and the model is linearized by the first-order Taylor expansion. Acarman et al. [8] approximate the hydraulic assist curve by using 8-degree polynomials and establish a 4-degree-of-freedom model of an online hydraulic steering system. Tai and Tomizuka [11] established a 2-degree-of-freedom linear model from motor torque to steering wheel

angle, steering wheel angle to wheel angle by linearizing the assist curve, and identified the unknown parameters by an open-loop frequency test. Garcia et al. [12] also established the state equation model of the self-driving steering system of commercial vehicles by using the linearized assist curve and the equivalent model of the mechanical system with 2-degree-of-freedom. Based on the above facts, the establishment of a 2-degree-of-freedom system model is sufficient for controller design.

As the low signal-to-noise ratio of the front wheel angle of the commercial vehicle, it can not be measured by the sensor, and the path tracking of the intelligent commercial vehicle is difficult to control the wheel directly. Therefore, the mainstream angle controller of EHCS uses the steering wheel angle instead of the wheel angle as the control target [13], and a large number of control methods have been proposed, such as fuzzy PID control [14], adaptive sliding control [15], and adaptive control [16, 17]. The method of a nonlinear robust controller is proposed by Tai and Tomizuka [11]; a first-order filter is introduced to reduce the inherent jitter of a nonlinear robust controller. Acarman et al. [8] designed a sliding mode controller to satisfy the robust stability for the modeling boundary and parameter uncertainty of the linear hydraulic steering system. Based on the linearized 2-degree-of-freedom model of the EHCS, Garcia et al. [12] designed the steering angle PI controller. Some simple controllers such as the PI controller are not easy to adjust the change in EHCS while artificial intelligence control with complex structure needs long adjustment time, which can not achieve an instantaneous dynamic response. Adaptive control can automatically compensate for unpredictable changes such as model parameters and input signals [18] matching the EHCS, so the model reference adaptive theory is used to design the angle controller of EHCS.

From the above discussions, the controller of EHCS based on the adaptive control method is developed. The 2-degree-of-freedom control model of EHCS is designed with reasonable assumptions and simplifies. Based on this model, the controller using model reference adaptive theory is designed by Lyapunov stability theory. Finally, the simulation results show that the system has good robustness under different load.

The main contributions of this paper are as follows. The cosimulation model and simplified 2-degree-of-freedom model are established for EHCS, which provides the basis for the development of the application of EHCS in the future. A controller based on model reference adaptive theory is proposed, which provides a new way to angle control of EHCS, and verifies the robustness of the controller under different load changes. It makes a certain contribution to the control of a steering system in an intelligent commercial vehicle.

The rest of the paper is organized as follows: Section 2 introduces the cosimulink model and simplified model. Section 3 designs a controller based on a parameter adaptive control system. Section 4 is the simulation results and analysis. Section 5 presents some conclusions.

2. Modeling of EHCS

The structure of EHCS is shown in Figure 1. The EHCS is divided into three parts: mechanical, hydraulic, and motor,

in which the mechanical model includes a steering wheel, torsion, worm gear, and circulating ball power steering gear. Hydraulic module contacts of a rotating valve, hydraulic circuit, and hydraulic pump. The mechanical and hydraulic models are built in AMEsim, and the torque control model of permanent magnet synchronous motor (PMSM) is built in Simulink.

2.1. Establish Cosimulation Model. The mechanical model and hydraulic model are established in AMEsim; the mechanical part is partially simplified as follows. The damping coefficient between worm gears and worms is ignored. The intelligent commercial vehicle is driverless, and the driver input torque is zero. The torsional stiffness of the upper torsion bar is represented by the rotating spring damper. The equivalent moment of inertia between the steering string and the worm gear is represented by the upper rotating load, while the moment of inertia between the worm wheel and the lower torsion bar is represented by the lower rotating load. The lower torsion bar module is equivalent to a rotating spring with stiffness of 1.5 Nm/deg, and the torsion bar deformation angle signal is collected by the steering sensor on the both ends of the spring. Motor driving torque is the only torque input of the model, which is built in Simulink. The circulating ball power steering gear is equivalent to a two-stage transmission pair containing a screw nut transmission pair and a rack tooth fan transmission pair [19]. The angle acquisition module is used to collect the rotation angle of the rocker shaft, ignoring the transmission ratio and elastic influence of the steering rod system. The rotation angle of the rocker shaft is regarded as the wheel angle.

The main component of the hydraulic module is the rotary valve, which is equivalent to a variable throttle according to its working principle [20], and the opening area of the variable throttle hole is controlled by the torsion bar deformation angle in the mechanical module. According to the parameters of the rotary valve and the method shown in the reference [19–21], the relationship between the overflow area of rotary valve port A and the deformation angle of the torsion bar θ_{lg} is calculated as shown in Equation (1). The hydraulic pump and piston adopt the module included in AMEsim.

$$\begin{cases} 7A + 85\theta_{\mathrm{lg}} - 180 = 0, & \theta_{\mathrm{lg}} \in [-5, -2], \\ 340A + 977\theta_{\mathrm{lg}} - 3486 = 0, & \theta_{\mathrm{lg}} \in (-2, 1.4], \\ 8295A - 13083\theta_{\mathrm{lg}} - 69994 = 0, & \theta_{\mathrm{lg}} \in (1.4, 5.35], \end{cases} \tag{1}$$

where A is the overflow area of the rotary valve and θ_{lg} is the deformation angle of the torsion bar.

The cosimulation part with Simulink is established, in which the input is the steering resistance moment of the front axle and the motor torque and the output is the rotation angle of the two wheels, the required speed of the motor and the steering wheel angle. The final model is shown in Figure 2.
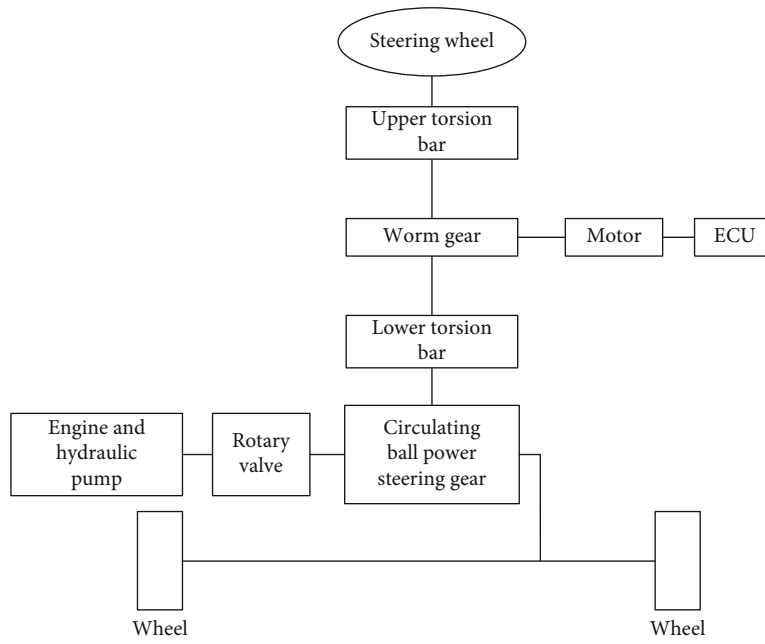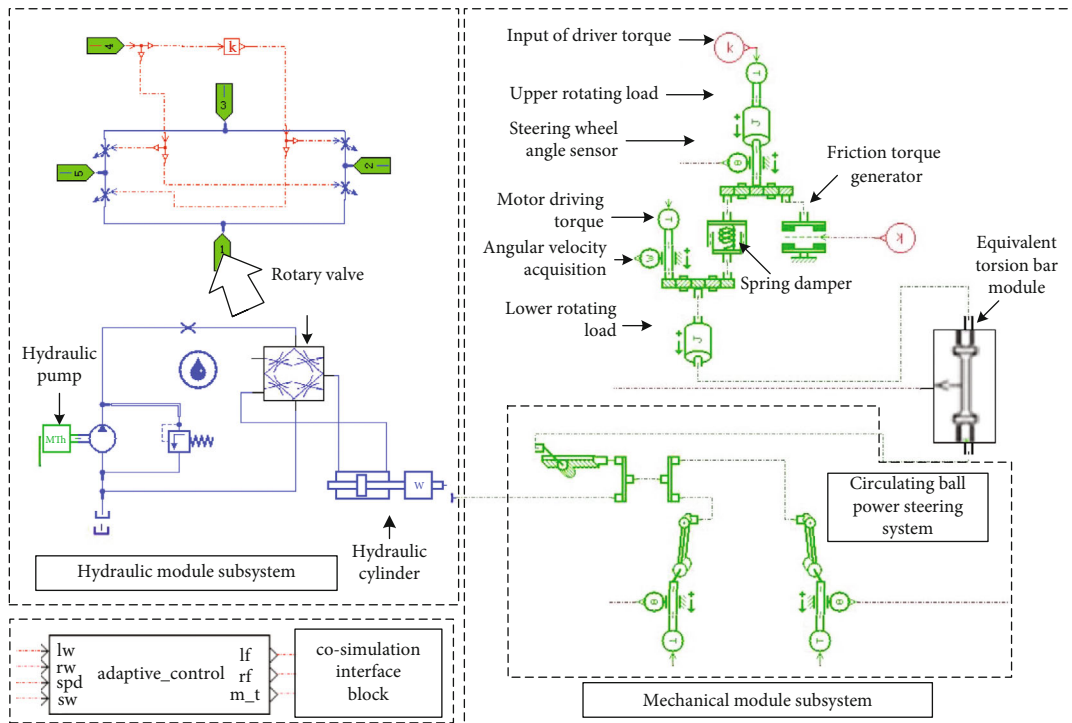
Figure 1: Structure diagram of EHCS.

Figure 2: Mechanical and hydraulic system modeling in AMEsim.

The dynamic model of the permanent magnet synchronous motor is shown in Equation (2). In order to facilitate control, the strong coupling current and voltage should be decoupled. If the d-axis current is controlled to zero, the motor torque is proportional to the q-axis current. As shown in Equation (3), the motor torque can be controlled by controlling the q-axis current.

$$T_e = p_n \left[\psi_f i_q + \left(L_d - L_q \right) i_q i_d \right],$$

$$T_e = J_e \frac{d\omega_m}{dt} + R_m \omega_m + T_L, \tag{2}$$
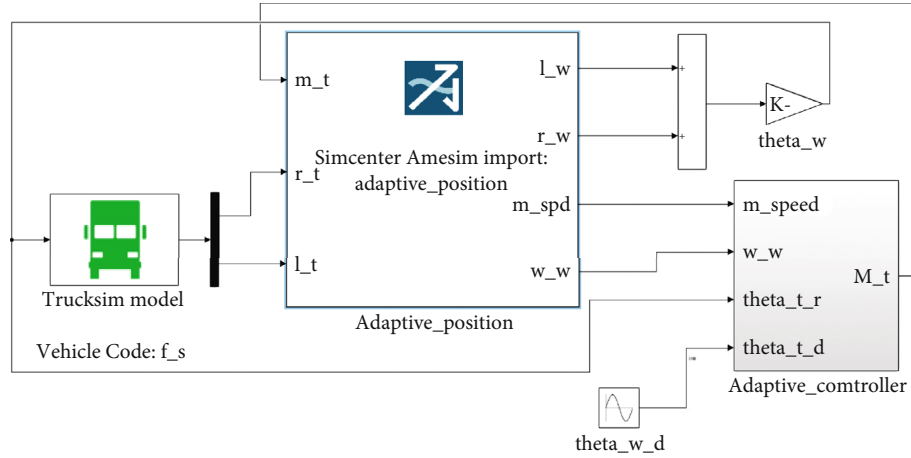
$$T_{e1} = p_n \psi_f i_q, \tag{3}$$

FIGURE 3: The combined simulation model.

where T_e and T_{e1} are the motor electromagnetic torque, T_L is the load torque, ψ_f is the stator winding flux, p_n is the motor pole logarithm, L_q and L_d are the q- and d-axis inductance, respectively, i_q and i_d are the q- and d-axis current, respectively, J_e is the motor moment of inertia, ω_m is the rotor mechanical angular speed, and R_m is the resistance coefficient.

The space vector pulse width modulation method, the inverter in Simulink, and the motor torque controller are used to build the torque control model of the permanent magnet synchronous motor (PMSM). The input of the model is the desired speed and q-axis current of the motor, and the output is the torque of the motor. In the TruckSim, a two-axle truck is selected, which is 5.5 t at no load and 9.5 t at full load. The transmission ratio of the steering system is set to 1, and the flexibility is set to 0. At this time, the wheel angle of the model in TruckSim is basically equal to the steering wheel angle. If the steering control of TruckSim is set to full, the wheel angle calculated in Simulink is input to the steering wheel angle in TruckSim, and the output of TruckSim is the steering resistance moment of the wheels on the front axle.

The cosimulation model is established as shown in Figure 3, including TruckSim function, AMEsim function, angle controller, and SVPWM motor controller.

2.2. Simplified Control Model. In order to simplify the EHCS as a 2-degree-of-freedom model, the following ignoring and assumptions are made: (1) inertia mass at the upper end of the lower torsion bar is equivalent to the worm gear, and inertia mass body at the lower end of the lower torsion bar is equivalent to the rocker shaft, dividing the whole steering system into two mass systems [20]; (2) the steering load is regarded as external interference; (3) the angle influence of the steering string is ignored; (4) the friction torque of the system is ignored; (5) the elastic influence of the steering rod system is ignored; (6) the elastic influence of the upper end of the upper torsion bar, that is, the steering string, is ignored. The force analysis of the simplified EHCS system is shown in Figure 4.

Ignoring the friction force of the system and linearizing the hydraulic assist force as $T_h = \psi_0 T_{k2}$, the dynamic model is shown in following:

$$J_u\ddot{\theta}_u + D_u\dot{\theta}_u + k_2\theta_u = T_u + nk_2\theta_d,$$
$$J_d\ddot{\theta}_d + D_d\dot{\theta}_d + \left(k_{rw} + n^2k_2 + n\psi_0k_2\right)\theta_d = (n + \psi_0)k_2\theta_u,$$

$$(4)$$

where J_u is the equivalent moment of inertia from the inertia mass at the top of the lower torsion bar to the worm gear, kg/m²; D_u is upper mass damping coefficient, Nm/rad/s; k_2 is the elastic stiffness of upper torsion bar, Nm/rad; J_d is the equivalent moment of inertia from the bottom of the lower torsion bar to the tooth fan shaft, kg/m²; D_d is lower mass damping coefficient, Nm/rad/s; k_{rw} is the simplified from steering system load to the stiffness of the spring model, Nm/rad; T_f is friction torque of lower mass body, Nm; n is the transmission ratio from the screw to the fan shaft, about 25; and T_h is the equivalent moment of the hydraulic system acting on the lower mass body, Nm.

Based on the above assumptions and ignoring, the steering wheel angle is approximately the worm gear angle, which is given as

$$\theta_s = \theta_u,$$
$$\theta_w = \theta_d,$$

$$(5)$$

where θ_s is the angle of the steering wheel and θ_{wh} is the angle of the wheel.

The transfer function of the controlled system can be obtained as shown as

$$G_p(S) = \frac{\theta_w(s)}{\theta_s(s)} = \frac{\theta_d(s)}{\theta_u(s)} = \frac{(n + \psi_0)k_2}{J_ds^2 + D_ds + (k_{rw} + n^2k_2 + n\psi_0k_2)}.$$

$$(6)$$

After standardizing the transfer function, it is concluded that the transfer function is in the overdamping state,
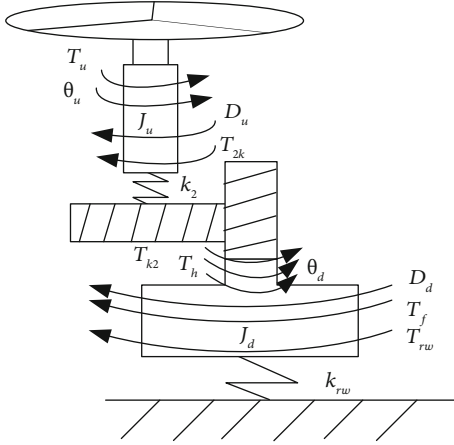
FIGURE 4: Diagram of force acting on a simplified system model.

meaning the response time is too long. In order to lay the groundwork for the model reference theory in the next chapter, an ideal transfer function model is proposed on the basis of the system transfer function. The following requirements are put forward for the ideal model: (1) It has a fast dynamic response and the rise time is less than 0.04 s. (2) The output amplitude should accurately express the input amplitude; in other words, the zero frequency amplitude should be 1. (3) The bandwidth of the system should meet the requirements of the steering system of the whole vehicle. (4) Within the required system bandwidth, the delay of the system should be less than 15 degrees as far as possible. The ideal model is a second-order model, and its transfer function is in a state of underdamping.

3. Angle Control Strategy Based on Model Reference Adaptive Theory

3.1. Design of Model Reference Adaptive Controller. Model reference adaptive control originates from the deterministic servo problem and is developed from the tracking problem. The system consists of a reference model, a controlled object, and a controller. The reference model is a given ideal model, which defines the performance indexes such as overshoot, damping performance, and adjustment time. The consistency between the controlled model and the reference model is measured by the state error vector [22]. The adaptive mechanism is designed according to the system stability criterion, and the corresponding adaptive law is designed to modify the variable parameters according to the direction of less deviation, so that the actual performance index of the system is close to the ideal performance index.

In this paper, aiming at the great variation of steering resistance moment with variable load of intelligent commercial vehicle and the nonlinear problems in the control of EHCS, a control framework is proposed based on model adaptive theory, and the control model is designed according to the ideal model as a reference model. Firstly, it is proposed that the state equation of the controlled system and the reference model with single input and single output is

shown as

$$
\begin{aligned}
\dot{x}_p &= A_p x_p + b_p u, \\
y_p &= h^T x_p, \\
\dot{x}_m &= A_m x_m + b_m u, \\
y_m &= h^T x_m,
\end{aligned}
\tag{7}
$$

where x_p is state vector of the controlled system, u is controlled vector, y_p is the output of a controlled system, x_m is the state vector of the reference model, and y_m is the output of the reference model.

The transfer functions of the controlled system and the reference model can be obtained from Equation (7) as shown as

$$
\begin{aligned}
G_p(s) &= h^T (sI - A_p)^{-1} b_p = k_p \frac{N_p(s)}{D_p(s)}, \\
G_m(s) &= h^T (sI - A_m)^{-1} b_m = k_m \frac{N_m(s)}{D_m(s)},
\end{aligned}
\tag{8}
$$

where k_p and k_m are gain coefficients.

The angle control of the EHCS takes the steering wheel angle as the input and the rocker shaft angle as the output. Define the error between the reference model and the actual output of the controlled system as $e_1 = y_m - y_p$, the control objectives of adaptive systems are shown as

$$
\lim_{t \to \infty} |e_1| = \lim_{t \to \infty} \left| y_m(t) - y_p(t) \right| = 0.
\tag{9}
$$

In order to make the adaptive adjustable system match the reference model completely, the adaptive system should have enough variable parameters, and the number of variable parameters should be the same as the uncertain parameters of the system. Define four variable parameters $\sigma = [\sigma_1 \quad \sigma_2 \quad \sigma_3 \quad \sigma_4]$ and two auxiliary signals in order to design controller, the auxiliary signal is shown as

$$
\begin{aligned}
\dot{\lambda}_1 &= A_f \lambda_1 + b_f u, \\
\dot{\lambda}_2 &= A_f \lambda_2 + b_f y_p,
\end{aligned}
\tag{10}
$$

where A_f and b_f are related to the $N_m(s)$ of the transfer function of the reference model.

Define w is the variable signal of the control system, $w = [r \quad -\lambda_1 \quad -y_p \quad -\lambda_2]$, the controller is built as shown in Figure 5.

$\hat{\sigma}$ is the variable parameter obtained in the ideal state, δ is the parameter error, and the augmented equation of state of the controlled system is obtained by substituting $\sigma = \hat{\sigma} + \delta$ into (7) as shown in (11). When $\delta = 0$, the augmented equation of state of the system in the ideal state, that is, the augmented equation of state of the reference model, is obtained as shown as
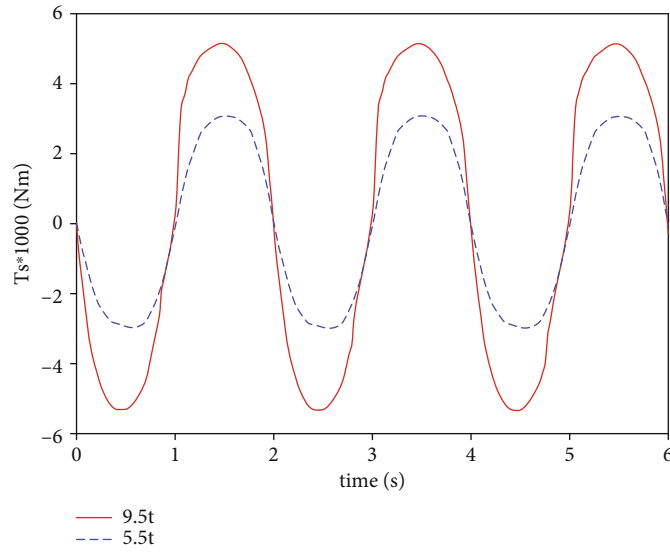
FIGURE 5: Control of model reference adaptive system.

FIGURE 6: The control action u and system output y_p.

$$\dot{x} = A_c x + b_c \left(\hat{\sigma}_1 r + \delta^T w \right),$$

$$\dot{x}_{mc} = A_{mc} x + b_{mc} \hat{\sigma}_1 r, \tag{11}$$

where $x^T = [x_p^T \quad \lambda_1^T \quad \lambda_2^T]$ and $x_{mc}^{\ T} = [x_{mp}^T \quad \lambda_{m1}^T \quad \lambda_{m2}^T]$.

The augmented error state equation can be obtained by subtraction of Equation (11) as given as

$$\dot{e} = A_c e + b_c \delta^T w = A_c e + b_c u_c, \tag{12}$$

where $e = [x_p - x_m, \lambda_1 - \lambda_{m1}, \lambda_2 - \lambda_{m2}]^T$ and $u_c = \delta^T w$.

3.2. Design of Adaptive Law according to Lyapunov Stability.

The adaptive law is designed by using Lyapunov stability theory, which is defined according to the Lyapunov function to satisfy the stability of the system [23], and the Lyapunov function and its derivatives are defined as shown as

$$V = \frac{1}{2} \left(e^T P e + \delta^T \Gamma^{-1} \delta \right),$$

$$\dot{V} = \frac{1}{2} e^T \left(P A_c + A_c^T P \right) e + \delta^T w b_c^T P e + + \delta^T \Gamma^{-1} \dot{\delta}$$

$$= \frac{1}{2} e^T \left(P A_c + A_c^T P \right) e + \delta^T \left(w b_c^T P e + \Gamma^{-1} \dot{\delta} \right) = \dot{V}_1 + \dot{V}_2.$$

$$(13)$$

It is obvious that the function V is positive definite, but in order to make the system stable, $\dot{V}$ should be negative definite.

According to Equation (12) and Kalman-Yakubovich theory [23], the transfer function $G_e(s)$ of e is strictly positive real if and only if there are positive definite matrices P and Q, such that Equation (14) holds. Therefore, when the transfer function of e is positive real, it can be obtained that $PA_c +$

FIGURE 7: Steering resistance moment.

(a) Variable parameters σ_1

(b) Variable parameters σ_2

(c) Variable parameters σ_3
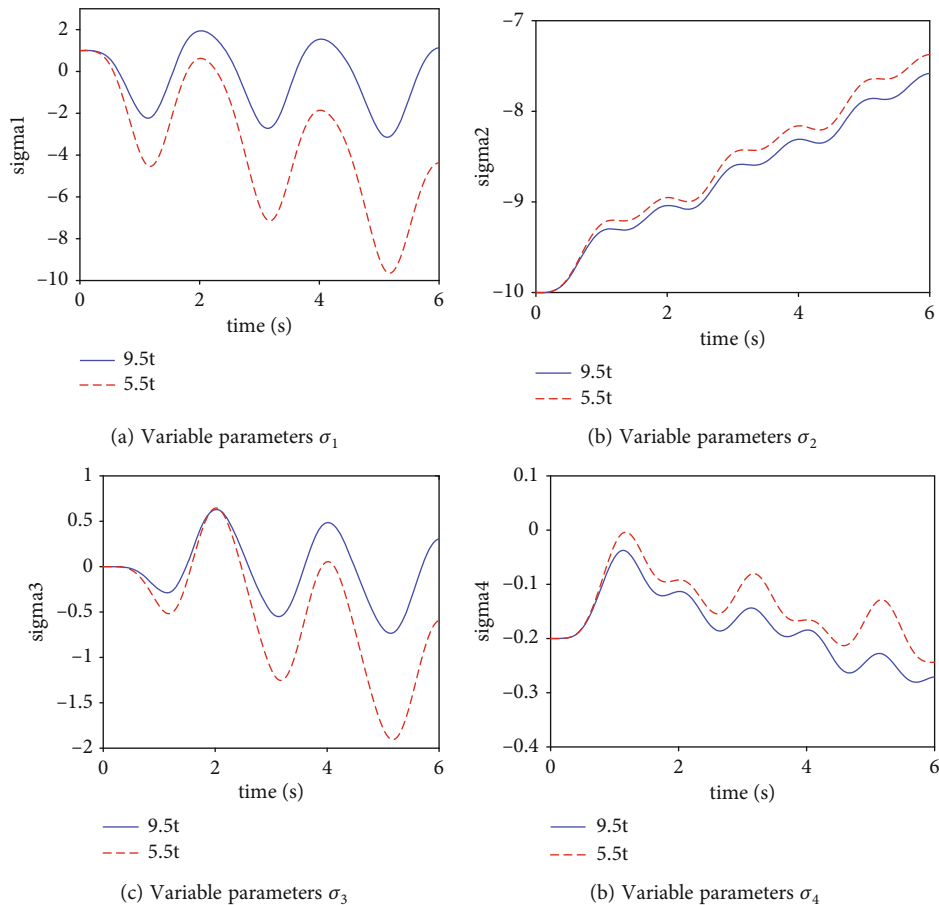
(b) Variable parameters σ_4

FIGURE 8: Comparison of variable parameters.

$A_c^T P = -Q$; by the words, $\dot{V}_1$ is negative definite.

$$A^T P + PA = -Q. \tag{14}$$

It is obviously to infer the $\dot{V}$ will be negative definite when $\dot{V}_2$ is zero. Hence, by defining adaptive law $\dot{\delta} = -\Gamma w\, b_c^T Pe$, $\dot{V}_2$ will be zero, which satisfies the condition that $\dot{V}$ is negative definite.

In order to make the strictly positive realness of the error model, the auxiliary polynomials $L(s) = s + a$ are introduced, and the transfer functions of the system model, the reference model, and the error model are shown as

$$
\begin{cases}
w_p(S) = L(s)G_p(s), \\
w_m(S) = L(s)G_m(s), \\
w_e(S) = L(s)G_e(s).
\end{cases}
\tag{15}
$$

Define $\zeta = L^{-1}(S)w = [r \quad -\lambda_1 \quad -y_p \quad -\lambda_2]^T/(s + a)$, the adaptive law of system variable parameters designed according to Lyapunov stability theory is shown as

$$
\begin{cases}
\dot{\sigma} = \dot{\hat{\sigma}} + \dot{\delta} = -\Gamma\zeta(t)e_1, \\
\dot{\zeta} = -a\zeta(t) + w(t),
\end{cases}
\tag{16}
$$

where Γ is a positive definite symmetric matrix.

The input of the controlled system is shown as

$$
u = \sigma^T w(t) + e(t)\zeta^T(t)\Gamma\zeta(t).
\tag{17}
$$

By introducing auxiliary polynomials $L(s) = s + 2$, it can be obviously proved that $w_p(S)$, $w_m(S)$, and $w_e(S)$ are strictly positive real. Define $\Gamma = \mathrm{diag}\ \{p_i\}$, $i = 1, 2, 3, 4$, according to the adaptive law of variable parameters as shown as

$$
\begin{aligned}
\dot{\sigma} &= -\Gamma L^{-1}(S)we_1(t), \\
\sigma &= -\int_0^t \Gamma\zeta(t)e_1(t)d\tau + \sigma(0).
\end{aligned}
\tag{18}
$$

The input of the controlled object is shown as

$$
\begin{aligned}
u &= \sigma^T(t)w(t) + e_1(t)\zeta^T(t)\Gamma\zeta(t) \\
&= \left[-\int_0^t \sum p_i\zeta_i e_1(t)d\tau + \sigma(0)\right]^T w(t) + e_1(t)\sum p_i\zeta_i^2.
\end{aligned}
\tag{19}
$$

The $\sigma(0)$ is the initial value of variable parameters; it can be optional in theory; however, a bad initial value may cause the system not to work normally. In addition, the selection of weighting matrix Γ has a great influence on the approximation reference model. Therefore, the selection of $\sigma(0)$ and Γ should be careful.

4. Robustness Verification of Angle Control Strategy

The cosimulation model in Section 2 is used to track the desired rotation angle of the front wheel under the load of 9.5 t and 5.5 t, respectively, and the sine waveform with a period of 2 s is selected to verify the simulation control strategy. The control action u and system output y_p under the two loads are shown in Figure 6, respectively. The steering resistance moment corresponding to the two loads is shown

in Figure 7, and the change of variable parameters during the operation of the model is shown in Figure 8.

It can be seen from the above facts that given the same desired rotation angle, the maximum difference of steering resistance moment under different loads is about 3000 Nm, but under the same expected rotation angle, the actual turning angle after the control strategy is basically the same. There is a certain oscillation in the first 0.3 s, which is caused by the same initial value of the variable parameters, and then, the system is stable after the variable parameters change with time. It is proved that the system has good robustness under the change of load.

5. Conclusion

Aiming at the EHCS with nonlinear uncertain parameters, after simplifying the control model to a 2-degree-of-freedom model, a controller based on adaptive theory is proposed by using Lyapunov stability theory. The adaptive law of variable parameters is designed, and the proposed control method can make the parameters change with time and compensate the nonlinear model disturbance. The load change is used to analyze the stability of control performance, and the results show that though the load difference is large, the operation of the EHCS can be effectively controlled and better performance can be obtained when tracking the target angle, meaning it has great robustness.

Data Availability

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

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] "Intelligent networked vehicle depth report: special topic on commercial vehicle applications," https://www.vzkoo.com/read/230df4ff483086ebe3528e669f8ad458.html?pid=1.

[2] G. Geng, H. Li, H. Jiang, J. Chen, and B. Tang, "Research on active return control of electro-hydraulic coupling steering system of commercial vehicle," *Automotive Engineering*, vol. 43, no. 6, pp. 899–908, 2021.

[3] Z. Tang, *Research on Path Tracking Control of Electro-Hydraulic Coupling Steering System for Commercial Vehicles [Master]*, Chongqing Jiaotong University, China, 2019.

[4] J. Sun, Y. Zhang, X. Shi, and J. Li, "Summary of the development of automobile power steering technology," *Heavy truck*, vol. 4, pp. 13-14, 2011.

[5] W. Cao, *Study on Characteristics and Lateral Motion Control of Electro-Hydraulic Coupled Power Steering System for Commercial Vehicles[Master]*, Yanshan University, China, 2020.

[6] K. Yu, "Development of electronically controlled hydraulic power steering system for NJ1062 freight car," *Journal of Nanjing Institute of Engineering*, vol. 1, no. 12, pp. 71–74, 2014.

[7] S. Wang, G. Shi, H. Zhang, C. Ju, and D. Sang, "Auto disturbance rejection control of steering wheel torque of electro-hydraulic coupling steering system," *Automotive Engineering*, vol. 43, no. 5, pp. 770–775+790, 2021.

[8] T. Acarman, K. A. Redmill, and U. Ozguner, "A robust controller design for drive by wire-hydraulic power steering system," in *Proceedings of the 2002 American Control Conference, Hilton Anchorage and Egan Convention Center*, Anchorage, Alaska, USA, 2002.

[9] C. Lv, *Research on Control Strategy of Electro-Hydraulic Coupling Steering System of Commercial Vehicle[Master]*, Jilin University, China, 2017.

[10] A. Dell'Amico, *On Electrohydraulic Pressure Control for Power Steering Applications*, 67 Linköping University, Sweden, 2016.

[11] M. Tai and M. Tomizuka, "Nonlinear robust loop shaping controller design for automated lane guidance of heavy vehicles," in *Proceedings of the American Control Conference*, pp. 2028–2032, Chicago, IL, USA, 2000.

[12] O. Garcıa, J. V. Ferreira, and A. de Miranda, "Dynamic model of a commercial vehicle for steering control and state estimation," in *XI Simposio Brasileiro de Automatice Inteligente 69 (SBAI)*, Chicago, IL, USA, 2013.

[13] W. Zhao, K. Gao, and W. Wang, "Prevention of instability control of commercial vehicle based on electric-hydraulic coupling steering system," *Journal of Jilin University (Engineering and Technology Edition)*, vol. 48, no. 5, pp. 1305–1312, 2018.

[14] J. Liu, Y. Liang, J. Chen, S. Wang, and B. Yan, "Research on brushless DC motor control system based on fuzzy PID control," *Journal of Physics: Conference Series*, vol. 1, no. 2033, pp. 1742–1750, 2021.

[15] C. Guan and S. Pan, "Adaptive sliding mode control of electro-hydraulic system with nonlinear unknown parameters," *Control Theory & Applications*, vol. 16, no. 11, pp. 1275–1284, 2008.

[16] Y. Li, R. Zhou, Q. Zheng, and L. Yang, "Design of model reference adaptive control law of steering mechanism for heavy duty vehicle," in *IEEE International Conference on Industrial Informatics*, pp. 347–352, Beijing, China, 2012.

[17] J. Zhao, Y. Lv, and Z. Zhao, "Adaptive learning based output-feedback optimal control of CT two-player zero-sum games," in *IEEE Transactions on Circuits and Systems II: Express Briefs*, 2021.

[18] W. Liu, L. Chen, and J. Chen, "Adaptive internal model control for automotive steer-by-wire system with time delay," *Journal of Shanghai Jiao Tong University*, vol. 55, no. 428, pp. 1210–1218, 2021.

[19] P. Pfeffer, M. Harrer, and D. Johnston, "Modelling of a hydraulic steering system," Fisita World Automotive Congress, 2006.

[20] Y. Chi, *Research on Lateral Control for Path Tracking of Commercial Vehicle Based on Electro-Hydraulic Coupled Steering[Master]*, Shandong University of Science and Technology, China, 2019.

[21] M. Tai, P. Hingwe, and M. Tomizuka, "Modeling and control of steering system of heavy vehicles for automated highway systems," *IEEE/ASME Transactions on Mechatronics*, vol. 9, no. 4, pp. 609–618, 2004.

[22] Y. Li and K. Zhang, *Adaptive Control Theory and Application*, Northwest University of Technology Press, Xi'an, China, 2005.

[23] H. Liao, C. Huang, and Y. Huang, "Control of AC servo motor based on Narendra stable adaptive controller," *Mechanical research and application*, vol. 27, no. 6, pp. 175–178, 2014.

[24] J. Zhao, J. Na, and G. Gao, "Adaptive dynamic programming based robust control of nonlinear systems with unmatched uncertainties," *Neurocomputing*, vol. 395, pp. 56–65, 2020.

Research Article

Wireless Sensor Network Target Localization Algorithm Based on Two- and Three-Dimensional Delaunay Partitions

Chenguang Shao ©

College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China

Correspondence should be addressed to Chenguang Shao; cgshao@nwnu.edu.cn

The target localization algorithm is critical in the field of wireless sensor networks (WSNs) and is widely used in many applications. In the conventional localization method, the location distribution of the anchor nodes is fixed and cannot be adjusted dynamically according to the deployment environment. The resulting localization accuracy is not high, and the localization algorithm is not applicable to three-dimensional (3D) conditions. Therefore, a Delaunay-triangulation-based WSN localization method, which can be adapted to two-dimensional (2D) and 3D conditions, was proposed. Based on the location of the target node, we searched for the triangle or tetrahedron surrounding the target node and designed the localization algorithm in stages to accurately calculate the coordinate value of the target. The relationship between the number of target nodes and the number of generated graphs was analysed through numerous experiments, and the proposed 2D localization algorithm was verified by extending it the 3D coordinate system. Experimental results revealed that the proposed algorithm can effectively improve the flexibility of the anchor node layout and target localization accuracy.

1. Introduction

With the rapid development of wireless communication technology and small embedded devices, WSNs have been used in numerous applications, such as target localization [1], environmental monitoring [2], smart factories [3], and agriculture and field habitat monitoring [4]. In practice, sensor network technology is used to monitor and collect data on specific targets in the area of interest.

The precise location coordinates of the target are critical for target monitors, and the data are analysed in data centres. For example, in sensitive areas involving rescue operations, the precise location of the target is first obtained. Typically, the location coordinates of the monitored target are unknown. Therefore, designing a scheme in which a WSN is used to accurately locate the target and constantly adjust the state when the location of the target moves is critical. Furthermore, the calculation efficiency of the proposed algorithm and the energy consumption of the nodes should be balanced. The accuracy of the positioning and comprehensive performance of the algorithm determines the quality of the localization method. GPS positioning systems can

achieve high accuracy, but the systems are not suitable for indoor environments and complex areas. When the location information of the target is determined using WSNs, the trajectory of the target can be estimated and drawn by combining related algorithms. The performance of positioning technology or methods markedly affects the localization accuracy and monitoring quality of WSNs.

Localization schemes are typically classified into distance-based localization schemes and distance-free localization schemes [5]. In distance-based localization scheme, the distance between the unknown node and the anchor node is estimated, whereas in the distance-free localization scheme, the location of the unknown node on the premise of uncertain distance is calculated [6]. Generally, the distance-based localization scheme exhibits a high localization accuracy, but its algorithm complexity is high, which requires computation. The distance-independent location scheme has low localization accuracy and low algorithm complexity. Range-based localization methods have been widely studied. The signal time of arrival (TOA) [7], time difference of arrival (TDOA) [8], angle of arrival (AOA) [9, 10], and received signal strength (RSS) are calculated.

The ranging method based on TOA technology is simple, but it requires considerable hardware performance, numerous computations, and high equipment costs. The TDOA method is an improvement of the TOA method and can accurately measure the coordinates of the target. However, this method requires two transmission signals of different rates, which results in a large operating overhead for the network. In addition, the AOA method must calculate the angle between the transmitting and receiving ends and must communicate and transmit in a line-of-sight environment. Therefore, the AOA is not suitable for precise locations in WSNs. The development of low-cost, fast calculation speed, and accurate localization methods is critical in WSN monitoring. Received signal strength indicator- (RSSI-) based localization methods have been widely used and commercialized [11–13]. The lost power between the transmitted and received power can thus be converted and calculated using a mathematical model. Furthermore, equipment hardware for WSN node localization using the RSSI method is commercialized and does not require additional separate components. RSSI-based localization technology is suitable for low-cost, high-precision, and large-scale WSNs.

The triangulation method has been studied in detail in fields, such as WSN coverage, routing algorithms, and localization, which optimise the node network layout and construct an optimal triangulation network. Therefore, the aim of this study is to optimise the location distribution of sensor nodes, improve the localization accuracy and performance of the localization algorithm, and extend it to the 3D coordinate system for testing.

(1) To solve the problems of layout optimization and low localization accuracy of anchor nodes occurring in conventional localization methods, the Delaunay triangulation method was introduced, and a localization scheme was designed to estimate the location coordinates of the target

(2) Based on the location of the target node, its surrounding triangle or tetrahedron were searched, and the location algorithm was designed in stages to accurately calculate the coordinate value of the target. The relationship between the number of target nodes and the number of generated graphs was analysed through numerous experiments, and the proposed 2D algorithm was verified by extending it to the 3D coordinate system

(3) We designed an experimental simulation and algorithm comparison and conducted numerous experiments with various parameters to verify the localization accuracy and reliability of the algorithm

The related work is reviewed in Section 2. In Section 3, the RSS channel model is presented, and related problems are described. The strategy we propose is detailed in Sections 4 and 5. In Sections 6 and 7, relevant experimental verification and algorithm comparison are designed. Concluding remarks alongside the future work are given in last section.

2. Related Works

The deployment method and algorithm design of anchor nodes are a research focus of target localization. In experiment tests performed by many researchers, the coordinate location distribution of anchor nodes generally occurs in the shape of a square, rectangle, and triangle [14–16]. Therefore, before experimentation, it is necessary to fix the location of the node in advance. If the location of the anchor node changed, the localization accuracy calculated also changed. Therefore, the fixed deployment mode of the anchor node is not suitable for the environment where nodes are randomly deployed.

Among the localization algorithm distance-based measurement, the most typical algorithms include the trilateral centroid localization, triangular measurement, least square method, and hyperbola localization algorithm. The trilateral centroid localization method has a small number of nodes and a low accuracy of target localization and is not suitable for 3D. The triangular measurement method needs to calculate the angle between the target and node, which undoubtedly increased cost. In [17], the author proposed a new sensor node localization scheme that improved the RSSI algorithm by considering power transmission and reception parameters to estimate the initial location of the node. A genetic algorithm is used to minimize the localization error, and the optimized coordinates are obtained by combining mutation and crossover operators. However, the layout optimization of the nodes is not considered, and the test environment is not suitable for 3D coordinates. In [18], two localization algorithms were designed based on anchor nodes, H-V scanning and diagonal localization algorithms, to estimate the coordinates of sensor nodes in the monitoring area. Among these algorithms, the diagonal localization algorithm belongs to the RSSI-based localization technology which can improve the localization accuracy of unknown nodes. However, the layout of anchor nodes is not sufficiently flexible to adjust dynamically according to the number of nodes, which may lead to nonsystematic errors. This method is suitable for 2D localization conditions but lacks a test of the real dimension. [19] proposed a robust ranging method to track the location of the target and used the trilateral localization method based on RSSI ranging. Consequently, the author reports the result of applying the trilateral localization technology to the measuring point and calculates the distance error between the ideal measuring point and measuring point through computer simulation. However, the computation speed of the trilateral localization method is slow, and the localization error is not accurate. In [20], the authors proposed a non-site-specific algorithm to better estimate the relationship between the RSS and distance. The author selected the most appropriate RSS value in the original RSS values through an algorithm to reduce the outlier effect, thereby ensuring the consistency between the RSS and distance relationship. In [21], the authors analysed the influence of two types of environmental interference on the RSSI value, used Kalman filter to preprocess the RSSI, and proposed a triangle centroid localization algorithm based on weighted feature points.

Experiments reveal that higher localization accuracy can be achieved, but its network did not have the characteristics of a self-organizing layout and is not suitable for 3D localization. In [14], to improve the precision of inside localization and optimize the allocation of node resources in WSNs, an equal-arc trilateral localization algorithm based on RSSI is proposed from the perspective of increasing measurement precision and bettering beacon nodes layout. Compared with the square layout, the triangle layout, and the improved triangle layout, the localization accuracy of this algorithm increased by 81%, 54%, and 48%, respectively. Finally, the author revealed that the proposed equal-arc triangulation algorithm can improve the localization accuracy and reasonably control sensor costs. However, in many applications, anchor nodes usually must be self-deployed for layout, and it is not possible to prespecify whether the layout of anchor nodes is square or other shapes. Delaunay triangulation has the optimal partition mode and geometric characteristics and is applied in the research direction of geometric routing, location, coverage, segmentation, data storage, and processing [22]. Li et al. [23] proposed a simple yet effective segmentation-based approach to detect trunk position and Delaunay triangulation (DT) geometry-based localization method for autonomous robots navigating in a forest environment. Experiments show the proposed method reach accurate global localization precision without a good initial pose or GPS signal.

The Delaunay division method can optimize the layout of 2D nodes, and applies to the network layout of 3D sensor nodes, based on the location coordinates of the nodes. Delaunay triangle segmentation can improve the flexibility of node layout and strengthen the correlation between anchor nodes. Therefore, we used the Delaunay partition method to optimize the Delaunay network layout for randomly deployed anchor nodes. Consequently, methods are designed to accurately estimate the coordinates of the target in 2D and 3D.

3. RSSI Ranging Principle and Problem Description

3.1. RSSI Ranging Principle. WSN localization methods are categorised into two ranging localization and localization without ranging depending on whether to measure the received signal and the transmitted signal is necessary. Localization technologies based on ranging primarily include TOA, TDOA, RSSI, and AOA. However, considering comprehensive indicators such as hardware cost, network computing power, and localization accuracy, the technical method based on RSSI is primarily used, combined with the improved localization algorithm to accurately locate the target. The primary principle of using RSSI ranging technology is used to establish a signal loss or attenuation model in the propagation process to estimate the distance between the transmitter and receiver. As displayed in Figure 1, the red line represents the waveform of the signal intensity varying with the distance under the ideal path loss model, and the blue line represents the waveform of the signal interference in the environment.

The localization algorithm based on the path loss model is used to determine the parameters of the path loss model according to the received RSSI data, and the model is used to estimate the distance value or further processing. We assume that the number of RSSIs received is M, and the RSSI value i received by node k is $\text{RSSI}(s, i)$[24]. Distance estimation was then performed based on the statistical RSSI measurement model.

$$\text{RSSI}_{(k,i)} = P_k - 10\eta_k \log(d_k) + v_{(k,i)}, \tag{1}$$

where d_k is the distance between the target and the anchor node k; P_k and η_k are the RSSI path loss model parameters of the anchor node k; and $v_{(k,i)}$ is a zero-mean Gaussian random distribution variable whose variance is equal to σ_k.

$$d_k = 10^{A_k - \text{RSSI}_k / 10\eta_k}. \tag{2}$$

The noise in the model obeys a Gaussian distribution, and the generated random variables are processed using the average value. When estimating the distance, the median value of a set of RSSI data is used for estimation $\overline{\text{RSSI}_k} = \text{RSSI}_k = \text{Median}\{\text{RSSI}_{(k,i)}, i = 1, \cdots, M\}$. The median value processing method was used to eliminate the random error of the original RSSI data, where $\overline{\text{RSSI}_k}$ represents the median value of the RSSI data collected by the anchor node k, which can be expressed as follows.

$$\overline{\text{RSSI}_k} = \frac{1}{M} \sum_{i=1}^{M} \text{RSSI}_{(k,i)}. \tag{3}$$

During propagation, the RSSI obeys the Gaussian distribution $N(\mu, \sigma^2)$ of the mathematical expectation μ and variance σ. The probability density function for any RSSI value is expressed as follows:

$$f(\overline{\text{RSSI}_k}) = \frac{1}{\sigma\sqrt{2\pi}} e^{-(\overline{\text{RSSI}_k} - \mu)/2\sigma^2}, \tag{4}$$

where the specific expressions of μ and σ are as follows:

$$\mu = \frac{1}{M} \sum_{i=1}^{M} \text{RSSI}_{(k,i)},$$
$$\sigma^2 = \frac{1}{M} \sum_{i=1}^{M} \left(\text{RSSI}_{(k,i)} - \mu\right)^2. \tag{5}$$

In the subsequent experimental verification process, after multiple sets of distance values measured by the anchor node, the localization method or algorithm is used to achieve an accurate estimation of the location of the target.

3.2. Problem Description. The method of using RSSI to determine the target distance is the first step in localization. Next, an efficient localization algorithm is designed to estimate the location coordinates of the target node. In the conventional method, a trilateral centroid localization algorithm with
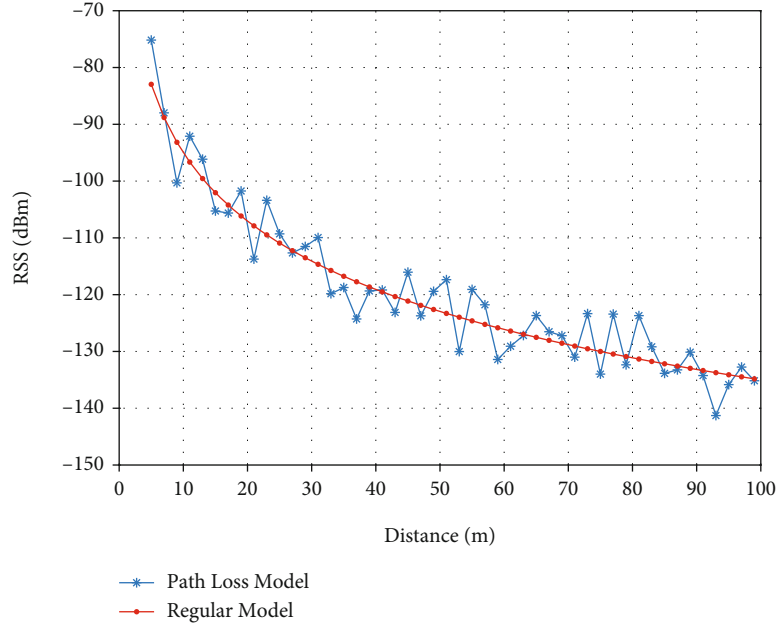
FIGURE 1: Schematic of the loss during RSS signal propagation.

low power consumption and fast measurement speed is used to solve the coordinates of the target. As displayed in Figure 2, the location coordinate of the node $T(x_t, y_t, z_t)$ to be tested is considered T, and the coordinates of the three anchor nodes, A, B, and C are $A(x_1, y_1, z_1)$, $B(x_2, y_2, z_2)$, and $C(x_3, y_3, z_3)$, respectively. Thus, the principle of trilateral measurement is that the distances between the three anchor nodes of A, B, and C to the target T are r_a, r_b, and r_c. Consequently, the circles formed by their respective measured radii intersect at a point T, and then, the location coordinates of the point T can be determined by establishing a system of equations.

However, in the test, the radii of points r_a, r_b, and r_c are r_a, r_b, and r_c, respectively, and their three circles cannot intersect at one point. As displayed in Figures 3 and 4, when the circles formed using the radii did not intersect at one point, Bulusu et al. [25] assumed $a(x_{a1}, y_{a1}, z_{z1})$, $b(x_{b1}, y_{b1}, z_{b1})$, and $c(x_{c1}, y_{c1}, z_{c1})$ as the intersection points formed by them. Next, determine the centre of mass of the three coordinates as the target coordinates. Thus, the approximate target coordinate value T can be obtained by calculation, and $T = ((x_{a1} + x_{b1} + x_{c1})/3, (x_{a2} + x_{b2} + x_{c2})/3, (x_{a3} + x_{b3} + x_{c3})/3)$ is calculated. However, the accuracy of the unknown target coordinates estimated by the trilateral centroid localization algorithm is not high, particularly when a certain height difference exists between the transmitter and receiver. In the calculation process for trilateral centroid localization, constraints should be considered, which is not conducive to large-scale network operations.

4. WSN Target Localization Algorithm Based on 2D Delaunay

4.1. Building a 2D Delaunay Network. In the Delaunay method, the region is divided according to the location of the node and a triangular network with an optimal layout is generated. According to the properties of Voronoi [26] and Delaunay [27], only the neighbouring nodes corresponding to the adjacent edges of Voronoi generate the corresponding triangle network, and the vertices of the triangle are composed of the nodes in the Voronoi unit body and its neighbouring nodes. In a nutshell, the vertices of each triangle in the Delaunay triangle network are composed of the three nearest nodes, and each side of the triangle will not intersect. Therefore, the construction of Delaunay triangulation will markedly reduce the localization time of anchor nodes and improve localization accuracy.

As shown in Figure 5, 70 anchor nodes s_i were randomly located on the plane L with an area of S_L, and the red dots represent the anchor nodes. Next, we used the Delaunay method to divide the area according to the 70 node coordinates $s_i(x_i, y_i, z_i)$. In Figure 6, the area divided by Delaunay is a triangular network composed of triangles N_Δ with various shapes. After the calculations are completed, in area L, the number of triangles N_Δ depends on the location coordinate $s_i(x_i, y_i, z_i)$ of the node and the number of nodes n. Thus, the greater the number of nodes is, the more triangles N_Δ are generated.

Initially, the number of randomly deployed nodes in area $L = 200 \times 200$ was n ($n = 100$), and the number of triangles in the Delaunay triangulation obtained from the test was $N_\Delta = 184$. The number of triangles was proportional to n, as listed in Table 1, which revealed the number of triangles generated by the Delaunay method when n has different values. Table 1 indicates that as the number of nodes increases, the number of triangles generated increases accordingly. Second, the total area S_0 of the formed Delaunay graph also changed with the number of nodes and location coordinates. Consequently, as the number of nodes n
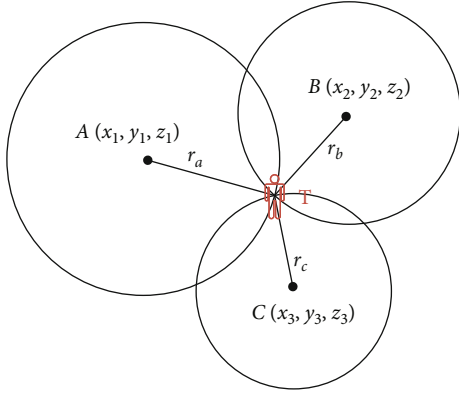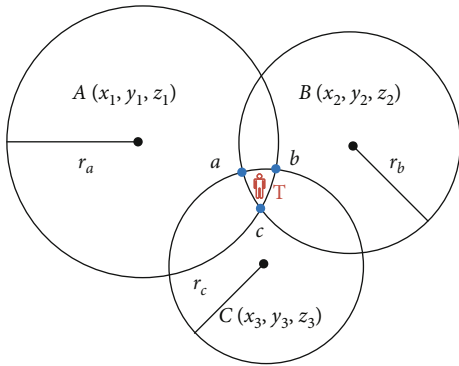
FIGURE 2: Three-sided positioning.
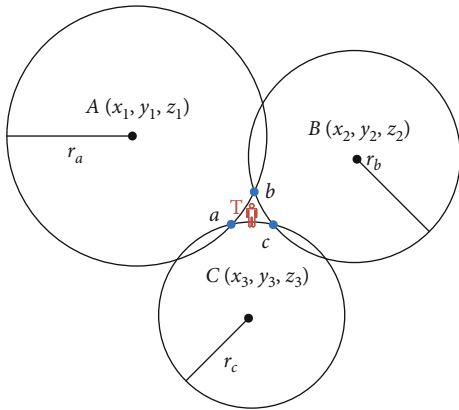

FIGURE 3: Intersect inside.

FIGURE 4: External intersection.

increased, the total area S_0 of Delaunay also increased, and its area ratio S_0/S_L gradually approached 100%.

4.2. Determining the Target Surrounded by Triangles. First, the coordinates of the nodes were divided using the Delaunay method, and the inner coordinates of each triangle were determined. The distance between the target and vertices of the triangle was calculated to determine the triangle surrounded by the target. The coordinates of the target are unknown; thus, the judgement method is the core step of positioning. In Figure 7, five black dots represent five nodes

with unknown coordinates. The coordinates of these five targets are: (127,135), (150,150), (145,170), (90,170), and (42,130). In a test, the coordinates of an unknown target cannot be determined in advance. Therefore, determining the triangle surrounded by the target is difficult. To address this problem, the area ratio method was used.

Figures 8 and 9, display a partial diagram of a certain part of the Delaunay triangulation. The coordinates of triangles and $\triangle ABC$ and $\triangle BCD$ are known and are $A(x_1, y_1)$, $B(x_2, y_2)$, $C(x_3, y_3)$, and $D(x_4, y_4)$, respectively. Because the coordinates of the target T are unknown, the distances between the four points A, B, C, D, and T can be measured by Equation (2) as d_{AT}, d_{BT}, d_{CT}, and d_{DT}, respectively. Therefore, determining whether T is in $\triangle ABC$ or inside the triangle $\triangle BCD$ is the key in this method.

If point T is inside the triangle $\triangle ABC$, then the area of the small triangle formed by T and $\triangle ABC$ satisfies the following conditions: $S_{\Delta ABT} + S_{\Delta ACT} + S_{\Delta BCT} \leq S_{\Delta ABC}$. If point T is not inside the triangle ΔABC, then it satisfies $S_{\Delta ABT} + S_{\Delta ACT} + S_{\Delta BCT} > S_{\Delta ABC}$. The area of $S_{\Delta ABC}$ can be calculated using Equation (8), and the area of the small triangle $S_{\Delta ABT}$, $S_{\Delta ACT}$, $S_{\Delta BCT}$ can be calculated using Helen formula as follows:

$$S = \frac{1}{4}\sqrt{[(d_1 + d_2 + d_3)(d_1 + d_2 - d_3)(d_1 + d_3 - d_2)(d_2 + d_3 - d_1)]},$$
(6)

where d_1, d_2, and d_3 are the distances between the vertices of targets T and ΔABC.

Thus, the target T can be estimated to be specifically located in a certain triangle in the Delaunay triangulation, and a corresponding localization method can be designed to calculate the coordinates of target T. Next, the subsequent localization method is implemented in stages in a real situation.

4.3. WSN Localization Algorithm Based on 2D Delaunay Partition. After this analysis, a target localization algorithm based on 2D Delaunay partitioning (2D-DPTL) was proposed, which is categorised into two stages to accurately locate the target.

Stage 1. When the number of anchor nodes is large, method 1 is used for localization.

Method 1. First, the number of known nodes n is randomly deployed in a plane with an area of L size. Then, the corresponding Delaunay triangulation is generated according to the position coordinates of the known node s_i (x_i, y_i) $(i = 1, 2, \cdots, n)$. Thus, in the first step, the judgement method described in the previous section is used to determine a triangle surrounded by T. Next, we calculated and generated the inner coordinates $s_j(x_j, y_j)$ corresponding to each Delaunay triangle. The coordinates of the inner point of the triangle can be calculated using the following expression:

$$s_j\left(x_j, y_j\right) = \left(\frac{(a_i x_i + b_i x_{i+1} + c_i x_{i+2})}{a_i + b_i + c_i)}, \frac{(a_i y_i + b_i y_{i+1} + c_i y_{i+2})}{(a_i + b_i + c_i)}\right),$$
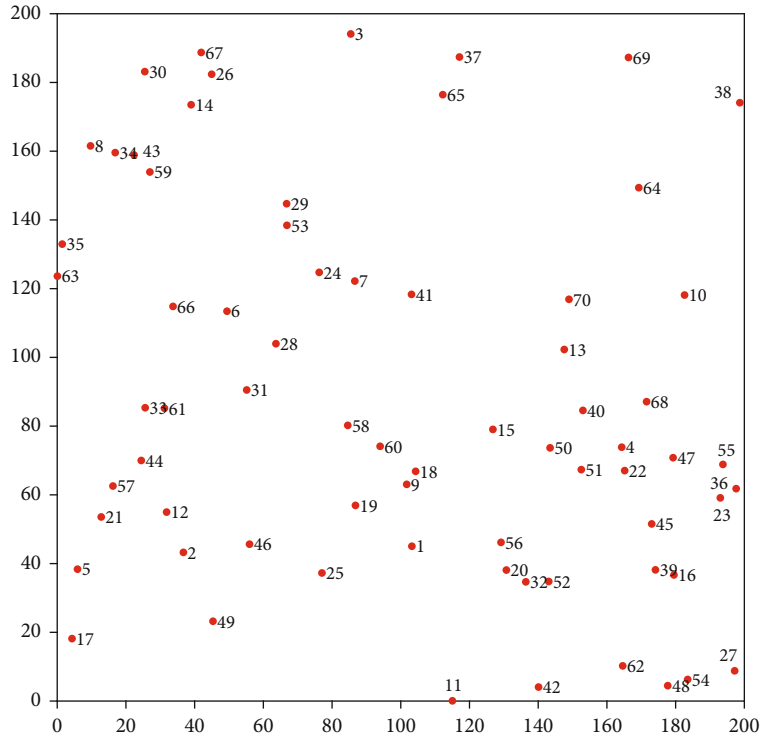(7)

Figure 5: Randomly generate anchor nodes.

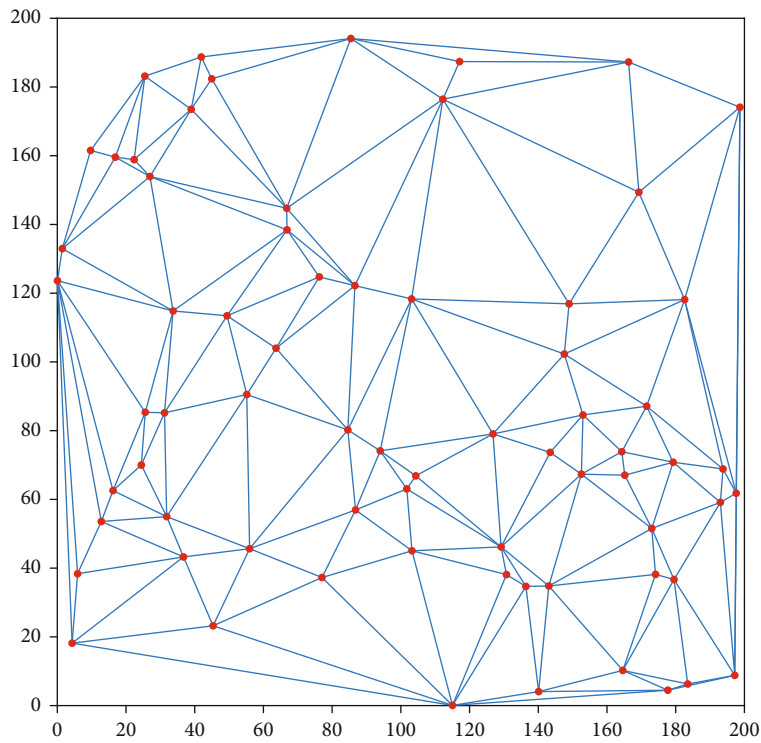Figure 6: Two-dimensional Delaunay division.

TABLE 1: Change in the number of anchor nodes.

L	n	$N_\triangle$	S_0 (m²)	S_0/S_L
200×200	30	50	29374	73.44%
200×200	50	89	34460	86.15%
200×200	70	126	30552	87.63%
200×200	100	184	36839	92.10%
200×200	150	283	36933	92.33%
200×200	200	383	37245	93.11%
200×200	300	581	37810	94.53%

where a_i, b_i, and c_i are the side lengths of the triangle and (x_i, y_i), (x_{i+1}, y_{i+1}), and (x_{i+2}, y_{i+2}) represent the vertices of the triangle corresponding to the inner centre.

As displayed in Figure 10, the blue five-pointed star represents the inner point of each triangle in the Delaunay triangle network, and its number is equal to the number $N_\triangle$ of triangles. The coordinates of each inner point are numbered, and the figure reveals that 126 inner points exist in total. Based on this method, the triangle surrounded by the target and its corresponding inner coordinates were determined. The red triangles in Figure 11 indicate the respective areas corresponding to the five target points. Next, the area ratio $S_k = S_j/S_m$ of the triangle to which the target $T_f(x_f, y_f)$ belongs is calculated. We assume that the area of all triangles is $S_j, j = 1, 2, \cdots, N_\triangle$, and the median $S_m = (S_{j+1}, j = 1, 2, \cdots, N_\triangle)/2$ or $S_m = (S_{(j+1)/2} + S_{(j)/2+1}, j = 1, 2, \cdots, N_\triangle)/2$ area of the triangle is obtained. If $S_j \le S_m$, then the inner coordinate $G_j(x_j, y_j)$ of the triangle is considered to be the estimated coordinate of the target $T_f(x_f, y_f)$ (i.e., $T_f(x_f, y_f) = s_j(x_j, y_j)$). If $S_j > S_m$, the centre of the triangle is used as the estimated value of T. However, this results in a large error. Thus, method 2 in stage 2 was used.

Stage 2. When the number of anchor nodes is small, method 2 is used for localization.

Method 2. When the number of anchor nodes is small, the number of triangles divided by Delaunay is small. Consequently, the area $S_j, j = 1, 2, \cdots, N_\triangle$ of the triangle where the target is located is too large, and the accuracy of using the inner coordinates in method 1 as target's estimated coordinates is too low. Therefore, in the first step, the method in phase 1 is used to determine the area ratio. Next, the calculated inner point G_j connects the vertex coordinates of the triangles A, B, and C to which it belongs and divides it into three small triangles, as illustrated in Figure 12, which are composed of green lines. We subsequently calculated the inner coordinates $G_1(x_4, y_4)$ of the small triangle where the target is located. As displayed in Figure 13, the three points A, B, and C represent a triangle formed in the Delaunay triangulation network, $G(x_0, y_0)$ represents its inner centre, and $G_1(x_4, y_4)$ represents the inner coordinate of the small triangle $\triangle BCG$. Thus, the coordinates of $G_1(x_4, y_4)$ can be selected as the estimated coordinates of the target point T (i.e., $G_1(x_4, y_4) = T_f(x_f, y_f)$), when the area ratio of the area

$(S_l, l = 1, 2, \cdots, N_L)$ of the small triangle $\triangle BCG$ to the area of S is $S_l/S \le 1$. For example, the area $\triangle BCG$ of a triangle $S_\triangle$ can be calculated using Equation (8), and $S_\triangle$ is a positive value (usually the absolute value $|S_\triangle|$).

$$S_\triangle = \frac{1}{2} \begin{vmatrix} x_1 & y_1 & 1 \\ x_2 & y_2 & 1 \\ x_3 & y_3 & 1 \end{vmatrix} = \frac{1}{2}(x_1 y_2 + x_2 y_3 + y_1 x_3 - x_3 y_2 - x_2 y_1 - x_1 y_3).$$

(8)

If the ratio of the area $(S_l, l = 1, 2, \cdots, N_L)$ of the small triangle $\triangle BCG$ where the target is located into the area of S_m satisfies the condition $S_l/S_m > 1$, the least squares method is used for calculation. We assumed that T is inside $\triangle BCG$ and used Equation (2) to measure the distance between each vertex of the small triangle $\triangle BCG$ and $T_f(x_f, y_f)$; thus, $d_1 = d_{T_1 G}$, $d_2 = d_{T_1 C}$, and $d_3 = d_{T_1 B}$. The coordinates of the three vertices of $\triangle BCG$ are $G(x_0, y_0)$, $B(x_3, y_3)$, and $C(x_2, y_2)$. Therefore, the following equations can be established:

$$\begin{cases} (x_0 - x_f)^2 + (y_0 - y_f)^2 = d_1, \\ (x_2 - x_f)^2 + (y_2 - y_f)^2 = d_2, \\ (x_3 - x_f)^2 + (y_3 - y_f)^2 = d_3. \end{cases}$$

(9)

Then, by subtracting d_3 from d_1 and d_2 in Equation (9), the following equation can be obtained:

$$\begin{cases} (x_0^2 - x_3^2) - 2(x_0 - x_3)x_f + y_0^2 - y_3^2 - 2y_f(y_0 - y_3) = d_1^2 - d_3^2, \\ (x_2^2 - x_3^2) - 2(x_2 - x_3)x_f + y_2^2 - y_3^2 - 2y_f(y_2 - y_3) = d_2^2 - d_3^2. \end{cases}$$

(10)

Equation ((11) can be expressed as a linear equation [28].

$$HX = b. \tag{11}$$

Among them,

$$H = \begin{bmatrix} 2(x_0 - x_3) & 2(y_0 - y_3) \\ 2(x_2 - x_3) & 2(y_2 - y_3) \end{bmatrix}, X = \begin{bmatrix} x_f \\ y_f \end{bmatrix}, \text{ and}$$

$$b = \begin{bmatrix} x_0^2 - x_3^2 + y_0^2 - y_3^2 + d_3^2 - d_1^2 \\ x_2^2 - x_3^2 + y_2^2 - y_3^2 + d_3^2 - d_2^2 \end{bmatrix}.$$

(12)

Thus, the coordinate of T can be solved by the following expression:

$$\tilde{X} = (H^T H)^{-1} H^T b. \tag{13}$$

However, the distance $d_1 = d_{T_1 G}$ from the target T to the inner heart $G(x_0, y_0)$ could not be accurately determined. The distance $d_{G_1 G}$ is approximately estimated as the distance of d_1,
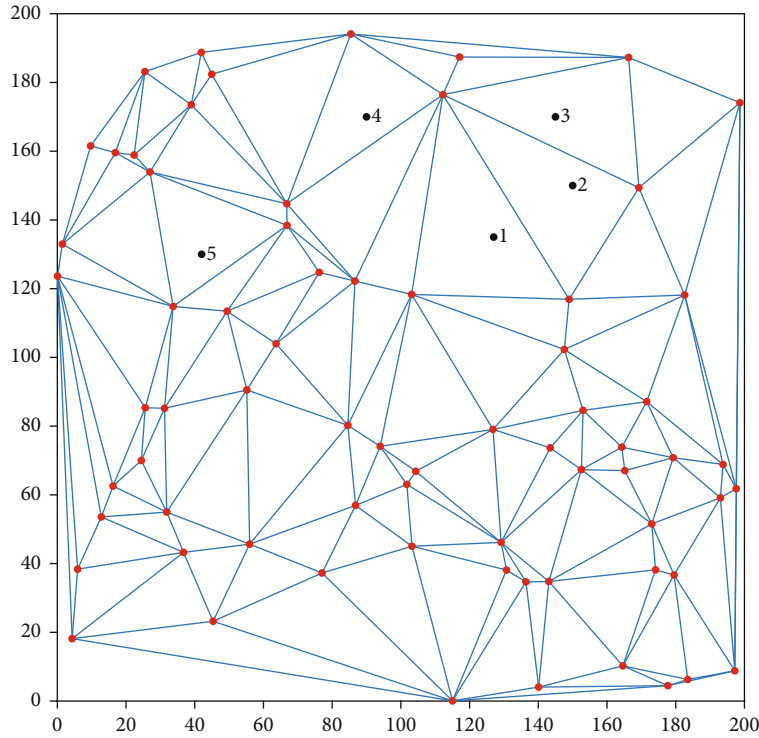
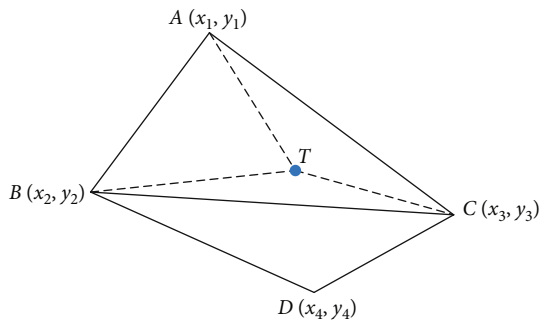FIGURE 7: Set the location of the target.
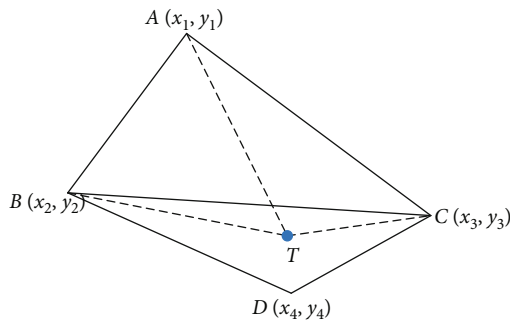

FIGURE 8: T is inside $\triangle ABC$.

FIGURE 9: T is outside $\triangle ABC$.

that is, $d_{G_1G} = d_1 = d_{T_1G}$. Among them, d_{G_1G} can be calculated using equation $d_{G_1G} = \sqrt{(x_0 - x_4)^2 + (y_0 - y_4)^2}$.

In general, methods 1 and 2 can be used in combination in a real localization process. The proposed 2D Delaunay target localization method can locate the target quickly according to the number of nodes.

5. WSN Target Localization Algorithm Based on 3D Delaunay

The proposed 2D WSN localization algorithm of the Delaunay division were extended to 3D. 3D localization algorithms are suitable for practical applications. The trilateral localization method can be used to determine the approximate coordinates of the target but cannot be used to measure the height of the z-axis. Therefore, referring to the localization method in Section 5.3, the 3D localization algorithm was designed according to various stages.

5.1. Three-Dimensional Space Divided by Delaunay. According to the content in Section 4, the Delaunay method is used to divide the coordinate points in the 2D plane, and the Delaunay figure obtained is a network composed of many triangles. When the Delaunay method is used to divide n the node 3D coordinates, the resulting graph is a 3D network composed of $N(N > n)$ tetrahedrons. As displayed in Figure 14, 100 anchor nodes are randomly deployed in a space of 200 m³, and the blue solid dots represent nodes with known coordinates. Next, the location coordinates of these
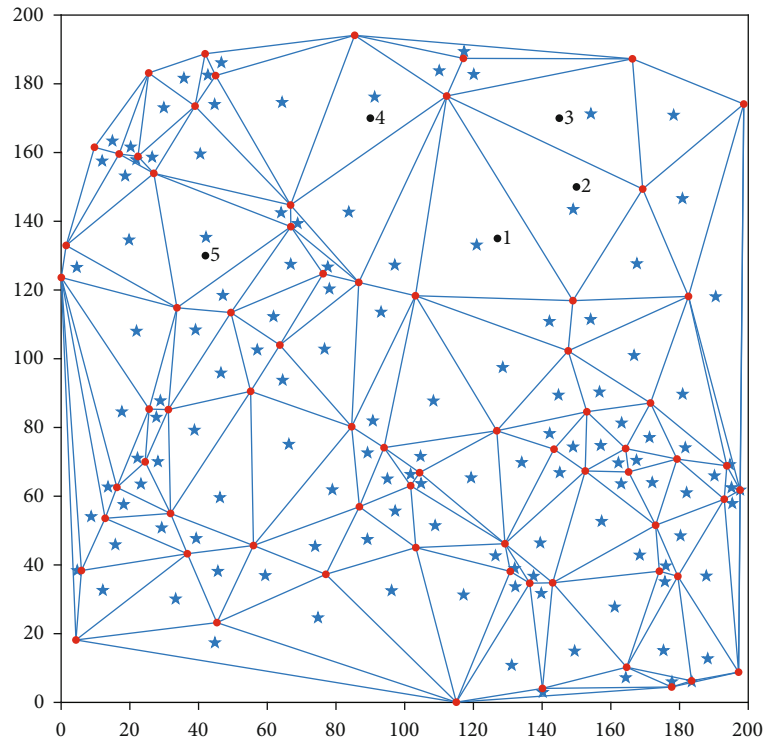
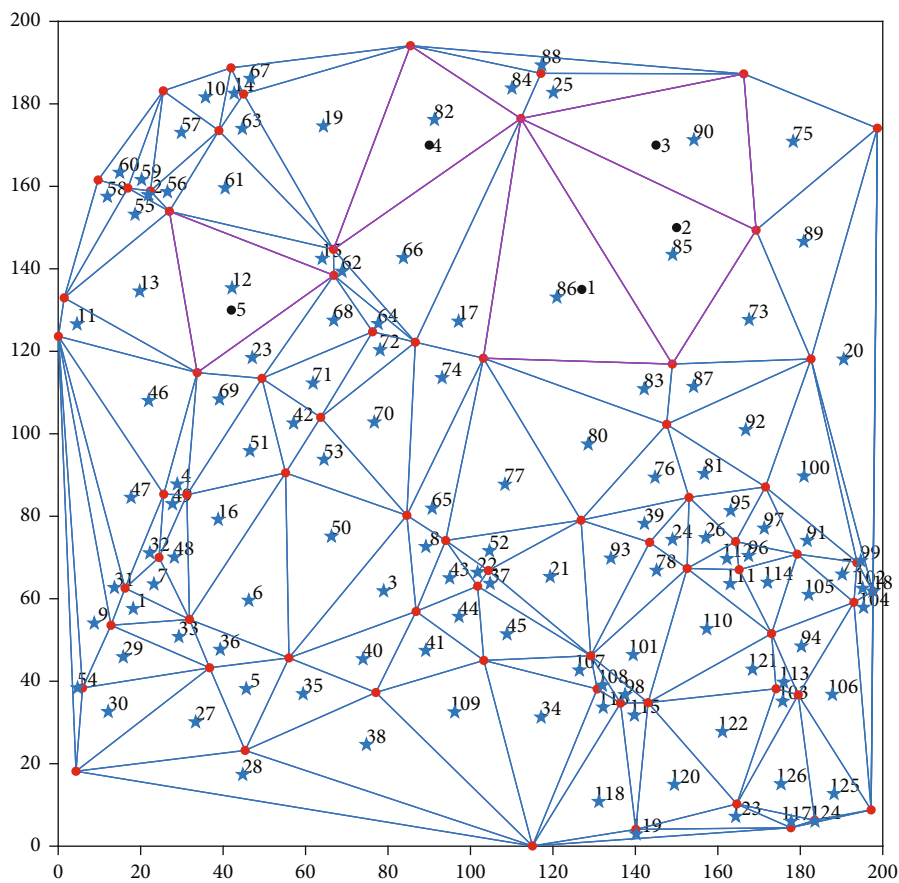FIGURE 10: Drawing the inner point of each triangle.

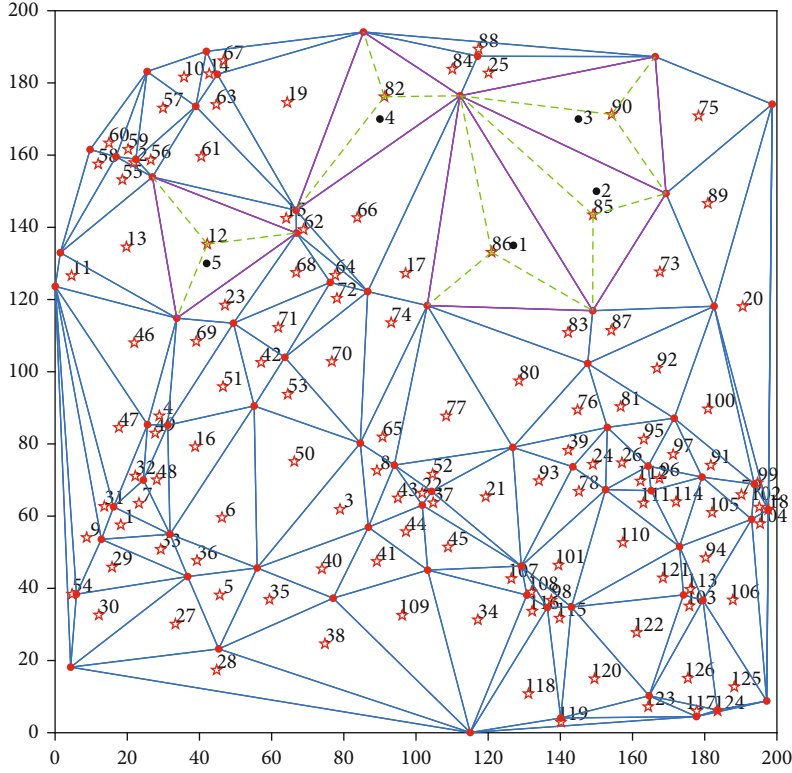FIGURE 11: Finding the triangle surrounded by the target.
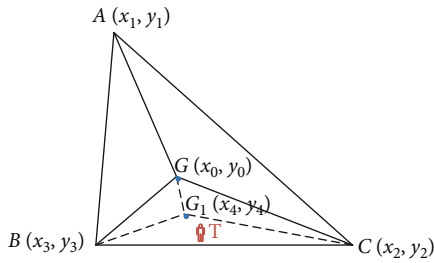
FIGURE 12: Delaunay second division.

FIGURE 13: Partial schematic.

100 nodes are divided using the Delaunay method, and the 12 edge lengths of cube 200 m³ were set as constrained edges. As displayed in Figure 15, the black 3D graphic network was composed of many tetrahedrons with different volumes. Comparing the graphs divided by the 2D Delaunay and 2D Delaunay methods reveals that the 2D Delaunay diagram is consisting of a triangular network composed of multiple small triangles, and the 3D Delaunay diagram is a 3D network consisting of multiple tetrahedrons. When the coordinate value of the node is fixed, the generated Delaunay network is unique.

The analysis revealed that the density of the triangular network or the 3D network after Delaunay is primarily determined by the number of nodes. Therefore, the data in Table 2 are used to verify the relationship between the 3D area size L, the number of nodes n, and the number of tetrahedrons N formed under the given conditions of the node coordinates.

Table 2 denotes that the greater the number of nodes is, the greater the number of tetrahedrons is. Thus, the total volume V of the formed Delaunay graph increases with an increased number of nodes. The number of inner points primarily depends on the number of nodes, and the size of the constraint space has limited effect on it. Many experiments have revealed that the size of V is primarily determined by the number of nodes and the location of nodes. Furthermore, the ratio of the total volume V of the tetrahedron formed by Delaunay to the volume V_0 of the deployment area increases as the number of nodes increases. However, the growth of the volume percentage V/V_0 was not considerably accelerated, and the volume ratio is low and between 50% and 70%. In Section 6, the distance between nodes is set through experiments to increase the volume ratio and optimise the location distribution of anchor nodes.

5.2. Determining Tetrahedron the Target Is In. The basic properties of Delaunay do not change with the increase in dimensionality, but the problems faced by the 3D localization method differ from those of 2D localization. When the coordinates of the nodes are divided using the Delaunay method, the inner centre of each tetrahedron should be calculated. Currently, estimating the location of the target T in the Delaunay tetrahedral network is the core step of 3D localization. The principle of the judgement strategy is similar to that of the 2D localization method in Section 4, but the method and algorithm to be calculated differ. Assuming that the vertices of each tetrahedron in the divided Delaunay
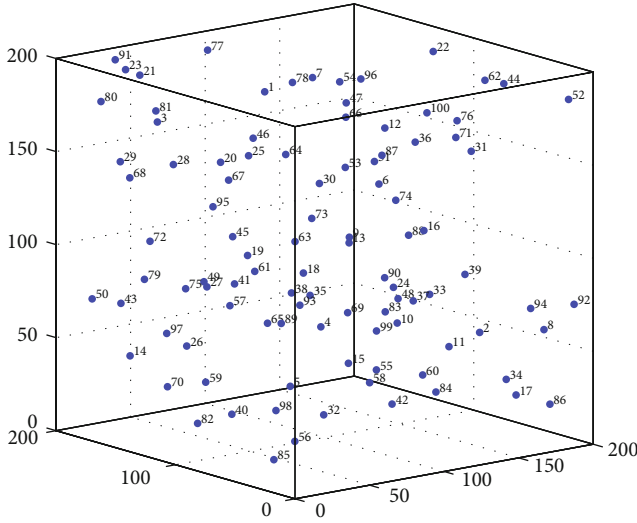
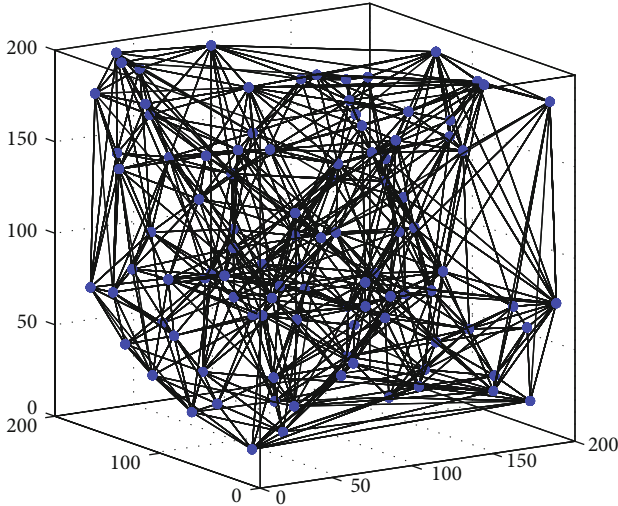Figure 14: Randomly deploy nodes.

Figure 15: Delaunay divides the 3D space.

are composed of known anchor nodes, using Equations (1)–(3), the RSSI values of the three transmission points with the strongest signal strength and the four closest distances $d_{4 \leq k \leq 6}$ between the known and unknown nodes can be calculated. Although the coordinates of the target are unknown, judging that T is located inside a certain tetrahedron is the first step in proposing a 3D WSN target localization algorithm.

As displayed in Figures 16 and 17, the tetrahedron $A - BCD$ represents a certain tetrahedron in the Delaunay 3D network, and $E - BCD$ represents the adjacent tetrahedron. Among these technologies, the coordinates of points A, B, C, D, and E are known and represent known nodes. The coordinate position of the unknown target T is uncertain, but the distance from the vertices A, B, C, D, and E of the tetrahedron can be measured by Equation (2) if d_{AT}, d_{BT}, d_{CT}, d_{DT}, and d_{ET} are assumed. If the point T is inside the tetrahedron $A - BCD$, the sum of the volume V of the tetrahe-

dron formed by the vertices of the target T and $A - BCD$ or T and $E - BCD$ satisfies the condition $V_{ABCT} + V_{ACDT} + V_{ABDT} + V_{BCDT} \leq V_{ABCD} V_{ABET} + V_{AECT} + V_{ABCT} + V_{BCET} \leq V_{ABCE}$. If the target T is not inside a tetrahedron $A - BCD$, then $V_{ABCT} + V_{ACDT} + V_{ABDT} + V_{BCDT} > V_{ABCD}$. To calculate the volume $A - BCD$ of the tetrahedron, the corresponding coordinate values are substituted into the Euler tetrahedron Formula (14).

$$V = \frac{1}{6} \begin{vmatrix} 1 & 1 & 1 & 1 \\ x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ z_1 & z_2 & z_3 & z_4 \end{vmatrix} = \frac{1}{6} \begin{vmatrix} x_2 - x_1 & y_2 - y_1 & z_2 - z_1 \\ x_3 - x_2 & y_3 - y_2 & z_3 - z_2 \\ x_4 - x_3 & y_4 - y_3 & z_4 - z_3 \end{vmatrix},$$

(14)

where x_i, y_i, and z_i are the known vertex coordinates of tetrahedron $A - BCD$ or $E - BCD$. The coordinates of the target T cannot be directly calculated, but the distance between each vertex of the tetrahedron can be estimated using the RSSI ranging model, which is set as d_1, d_2, d_3, d_4, and d_5. Therefore, tetrahedron's edge length formula can be used to calculate the volume, and Equation (15) can be used to calculate the volume V_l of the small tetrahedrons V_{ABCT}, V_{ACDT}, V_{ABDT}, and V_{BCDT}.

$$V_l^2 = \frac{1}{288} \begin{vmatrix} 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & d_1^2 & d_2^2 & d_3^2 \\ 1 & d_1^2 & 0 & d_4^2 & d_5^2 \\ 1 & d_2^2 & d_4^2 & 0 & d_6^2 \\ 1 & d_3^2 & d_5^2 & d_6^2 & 0 \end{vmatrix}.$$

(15)

Among these values, V_l is a positive value. Therefore, the corresponding tetrahedral volume can be calculated using Equations (14) and (15) as follows: using the aforementioned method, we determined the tetrahedron in the Delaunay network target is inside.

5.3. 3D WSN Target Localization Algorithm Based on 3D Delaunay. A novel WSN target localization algorithm based on 3D Delaunay was proposed. According to the relationship between the number of nodes and the number of tetrahedrons after the Delaunay division, the algorithm was designed in two stages. In the first stage, the number of nodes and area range were determined. The number of tetrahedrons generated by the 3D Delaunay method was approximately three times greater than the number formed by the 2D Delaunay method. Therefore, the 3D Delaunay method was more suitable for target localization with many nodes. In the second stage, the number of nodes is small, and the analysis and design are performed with reference to the 2D Delaunay localization method.

TABLE 2: Verifying the relationship between the number of nodes and other variables.

L	n	N	V (m^3)	V/V_0	L	n	N	V (m^3)	V/V_0
$200 \times 200 \times 200$	100	512	$5.3744e+06$	67.18%	$200 \times 200 \times 20$	100	483	$5.3744e+05$	67.18%
$200 \times 200 \times 200$	70	327	$4.8725e+06$	60.91%	$200 \times 200 \times 20$	70	319	$4.8725e+05$	60.91%
$200 \times 200 \times 200$	50	228	$4.3816e+06$	54.77%	$200 \times 200 \times 20$	50	211	$4.3816e+05$	54.77%
$200 \times 200 \times 200$	20	61	$2.2605e+06$	28.26%	$100 \times 100 \times 20$	70	323	$1.2181e+05$	60.90%
$200 \times 200 \times 50$	100	479	$1.3436e+06$	67.18%	$100 \times 100 \times 20$	50	214	$1.0954e+05$	54.77%
$200 \times 200 \times 50$	70	333	$1.2181e+06$	60.90%	$100 \times 50 \times 20$	70	319	$6.0906e+04$	60.91%
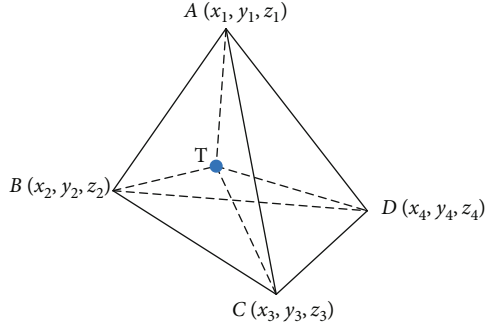
FIGURE 16: Point T is inside $A - BCD$.
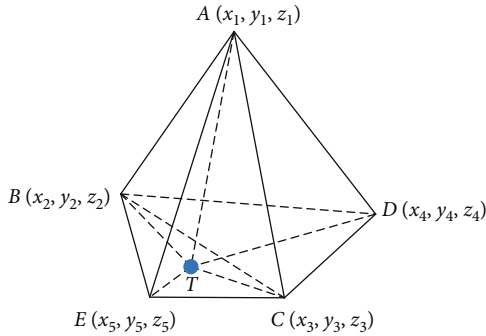
FIGURE 17: Point T is outside $A - BCD$.
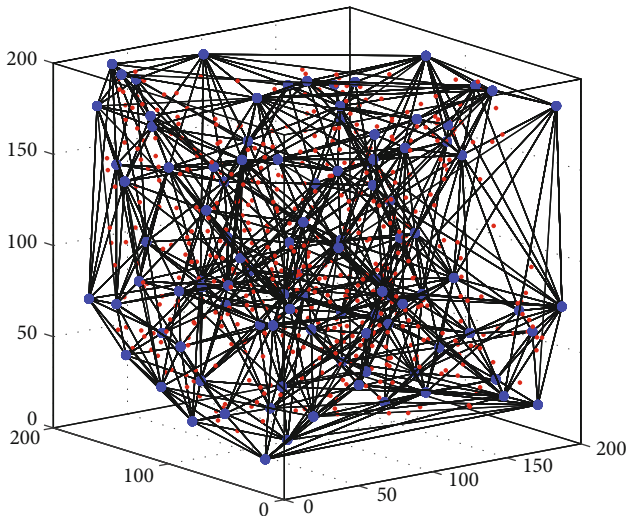
FIGURE 18: Calculating the inner point of a tetrahedron.

Stage 1. When the number of nodes is large, the coordinates of the centre I of the tetrahedron are mainly used to estimate the coordinates of the target.

Method 1. First, we calculated the inner coordinates $I(x_i, y_i, z_i)$ of each tetrahedron. In Figure 18, the red dots represent the inner points of each tetrahedron, the number of which is equal to the number of tetrahedrons. According to the method in Section 5.2, we find the tetrahedron surrounded by the target T and use Equation (16) to calculate the inner coordinates $I(x_i, y_i, z_i)$ of the tetrahedron.

$$
\begin{cases}
x_1 = \dfrac{S_1 x_1 + S_2 x_2 + S_3 x_3 + S_4 x_4}{S_1 + S_2 + S_3 + S_4}, \\
y_1 = \dfrac{S_1 y_1 + S_2 y_2 + S_3 y_3 + S_4 y_4}{S_1 + S_2 + S_3 + S_4}, \\
z_1 = \dfrac{S_1 z_1 + S_2 z_2 + S_3 z_3 + S_4 z_4}{S_1 + S_2 + S_3 + S_4}.
\end{cases}
\tag{16}
$$

Assume that the vertex coordinates of the tetrahedron are (x_1, y_1, z_1), (x_2, y_2, z_2), (x_3, y_3, z_3), and (x_4, y_4, z_4). Here, $S_i (i = 1, 2, 3, 4)$ represents the side area of the tetrahedron, and the inner point $I(x_i, y_i, z_i)$ of the tetrahedron can be calculated using Equation (16). In the experiment, three unknown target coordinates are given. Next, the tetrahedron where the target is located is determined by using the method described in Section 5.2. As displayed in Figure 19, the three black tetrahedrons each contain three targets.

To avoid selecting the inner points of some tetrahedrons that are too large, the coordinates of the target are assumed to be inaccurate. Therefore, the volume ratio of the tetrahedron was determined to solve the afore-mentioned problem. The volume of all tetrahedrons is V_j ($j = 1, 2, \cdots, N_p$), where N_p is the number of tetrahedrons. Next, the volume median $V_m = ((V_{j+1})/2, j = 1, 3, \cdots, N_p)$ or $V_m = (S_{(j+1)/2} + S_{(j)/2})/2, j = 2, 4, \cdots, N_p)$ of the divided tetrahedron was calculated. Assuming that the volume of the tetrahedron in which the target T is located satisfies $V_j \leq V_m$, the tetrahedral centre coordinate $G_j(x_j, y_j)$ can be replaced with the coordinate of the target T, that is, $F_f(x_f, y_f) = I_j(x_j, y_j)$. Assuming that $V_j > V_m$, the method in phase 2 is used.

Stage 2. When fewer known node coordinates exist, and the formed tetrahedron volume satisfies the condition
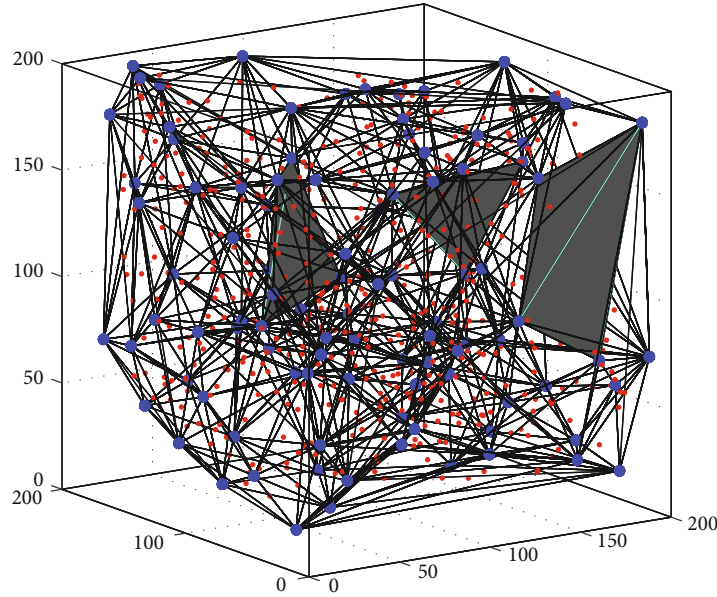
FIGURE 19: Tetrahedron surrounded by target.

$V_j > V_m$, the inner coordinates cannot be estimated as the coordinates of the unknown target.

Method 2. When $V_j > V_m$ is determined in the first step, the coordinates of the target T are directly solved using the least squares method. As displayed in Figures 20 and 21, the red mark indicates target T, and T is inside the $A - BCD$ tetrahedron. When the tetrahedron surrounded by the target T is located for the first time, the least square method combined with the RSSI ranging principle is used to solve the problem with less computation time.

Assuming that T is inside $A - BCD$, we use Equation (2) to measure the distance between the four vertices of the tetrahedron $A - BCD$ and $T_f(x_f, y_f)$, where $d_1 = AT$, $d_2 = BT$, $d_3 = CT$, and $d_4 = DT$. Consequently, the coordinates of the three vertices of $\triangle BCG$ are known, $A(x_1, y_1, z_1)$, $B(x_2,$ $y_2, z_2)$, $C(x_3, y_3, z_3)$, and $D(x_4, y_4, z_4)$, which can be calculated as follows:

$$\begin{cases} (x_1 - x_f)^2 + (y_1 - y_f)^2 + (z_1 - z_f)^2 = d_1, \\ (x_2 - x_f)^2 + (y_2 - y_f)^2 + (z_2 - z_f)^2 = d_2, \\ (x_3 - x_f)^2 + (y_3 - y_f)^2 + (z_3 - z_f)^2 = d_3, \\ (x_4 - x_f)^2 + (y_4 - y_f)^2 + (z_4 - z_f)^2 = d_4. \end{cases} \tag{17}$$

Next, subtract d_4 from d_1, d_2, and d_3 from Equation (17) to obtain the following expression:

$$\begin{cases} (x_1^2 - x_4^2) - 2(x_1 - x_4)x_f + (y_1^2 - y_4^2) - 2(y_1 - y_4)y_f + (z_1^2 - z_4^2) - 2(z_1 - z_4)z_f = d_1^2 - d_4^2, \\ (x_2^2 - x_4^2) - 2(x_2 - x_4)x_f + (y_2^2 - y_4^2) - 2(y_2 - y_4)y_f + (z_2^2 - z_4^2) - 2(z_2 - z_4)z_f = d_2^2 - d_4^2, \\ (x_3^2 - x_4^2) - 2(x_3 - x_4)x_f + (y_3^2 - y_4^2) - 2(y_3 - y_4)y_f + (z_3^2 - z_4^2) - 2(z_3 - z_4)z_f = d_3^2 - d_4^2. \end{cases} \tag{18}$$

Next, Equation (18) is rewritten as a system of linear equations, similar to Equation (11), where

$$H = \begin{bmatrix} 2(x_1 - x_4) & 2(y_1 - y_4) & 2(z_1 - z_4) \\ 2(x_2 - x_4) & 2(x_2 - x_4) & 2(z_2 - z_4) \\ 2(x_3 - x_4) & 2(y_3 - y_4) & 2(z_3 - z_4) \end{bmatrix}, X = \begin{bmatrix} x_f \\ y_f \\ z_f \end{bmatrix}, \tag{19}$$

and b can be rewritten as a determinant

$$b = \begin{bmatrix} x_1^2 - x_4^2 + y_1^2 - y_4^2 + z_1^2 - z_4^2 = d_4^2 - d_1^2 \\ x_2^2 - x_4^2 + y_2^2 - y_4^2 + z_2^2 - z_4^2 = d_4^2 - d_2^2 \\ x_3^2 - x_4^2 + y_3^2 - y_4^2 + z_3^2 - z_4^2 = d_4^2 - d_3^2 \end{bmatrix}. \tag{20}$$

Thus, Equation (9) is used to calculate the determinant coordinate value of the target T, and then the corresponding value is removed.
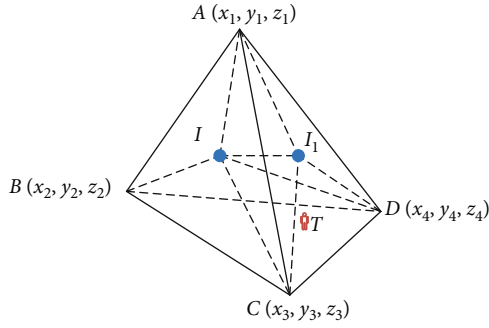
FIGURE 20: T is inside the tetrahedron.
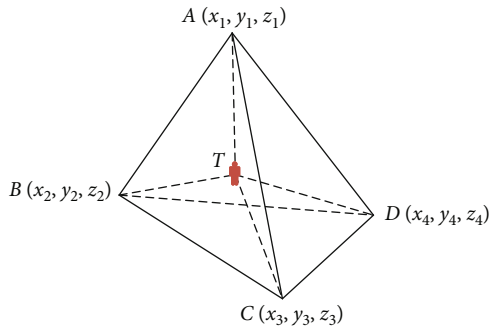
FIGURE 21: Distance from T to the tetrahedron vertex.

6. Experimental Verification and Result Analysis

6.1. A Linear Regression Analysis of Relevant Data. Experiment 1. Determining the relationship between the number of anchor nodes and the number of graphs generated by the Delaunay division.

Comparing the graphs formed by the 2D and 3D Delaunay methods revealed that the 2D Delaunay graph is a triangular network composed of multiple small triangles, while the 3D Delaunay graph is a 3D network composed of multiple tetrahedrons. When changing L, the number of tetrahedrons primarily depends on the number of nodes n. The relationship between the number of nodes divided by the 2D and 3D Delaunay methods and the number of corresponding graphics were analysed. With the same number of nodes, the number of triangles generated by the Delaunay method is considerably smaller than the number of tetrahedrons. Thus, the number of graphs generated by the Delaunay method is linearly related to the number of nodes. As displayed in Figure 22, we set the size of the 2D area to $L = 200\,\mathrm{m}^2$ and the 3D area to $L = 200\,\mathrm{m}^3$. Next, we set the number of different nodes $n(20 \leq n \leq 200)$, obtain the number of triangles N_Δ and the number of tetrahedrons N, and draw a scatter plot of the two. Figure 22 reveals that the number of triangles or tetrahedrons formed after the Delaunay division appears to have a linear relationship with the number of nodes n. To verify this result, linear regression analysis was performed using Excel software.
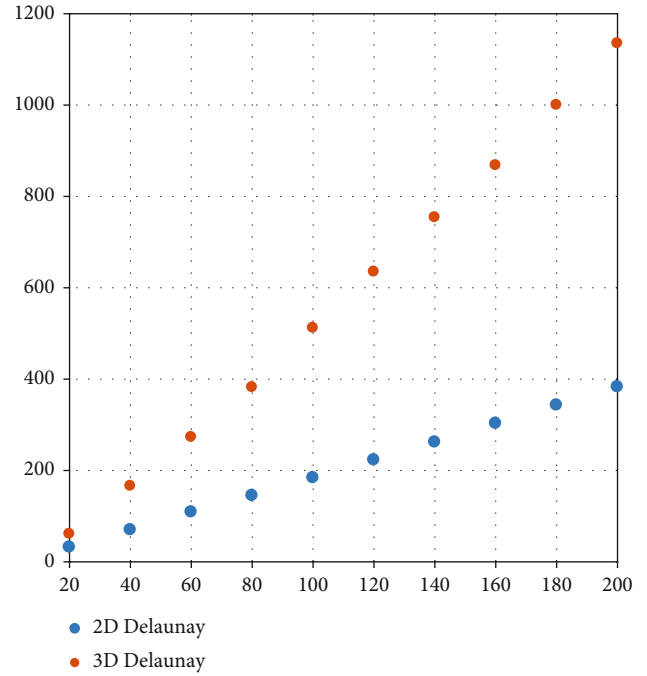

FIGURE 22: Delaunay scatter chart.

Assume that the linear equations of two of them can be expressed as follows:

$$\begin{cases} y_1 = ax_0 + b, & (2\mathrm{D}), \\ y_2 = cx_1 + d, & (3\mathrm{D}). \end{cases} \tag{21}$$

When using Excel software to analyse its linear regression characteristics, select the confidence area as 95% and the constant as 0, and generate the corresponding predictive regression and residual plots. Next, the Delaunay linear regression equation and regression characteristics of 2D and 3D data were calculated, and the linear regression equation fitting diagram shown in Figure 23 was obtained.

Figure 23 displays the linear regression equation $y_1 = 1.887x$. Among these results, the coefficient of determination $R^2 = 1$ of the equation y_1 indicates that the data fit is excellent and close to 1. The value obtained by the regression analysis satisfies the condition F value $= P$ value < 0.01, which indicates that the regression effect of the linear equation is significant. Similarly, to analyse the data obtained by the 3D Delaunay division, the equation is $y_2 = 5.4132x$. Furthermore, all the indicators satisfy the linear regression. Therefore, the number of triangles or tetrahedrons obtained after Delaunay division is positively linearly related to the number of nodes.

Finally, residual analysis was used for the 2D and 3D data, as displayed in the residual diagrams in Figures 24 and 25. In the figures, the two sets of data are evenly distributed on both sides of the symmetry axis $x = 0$, which demonstrates that the variables are linearly distributed. Finally, through afore-mentioned analysis, the number of triangles
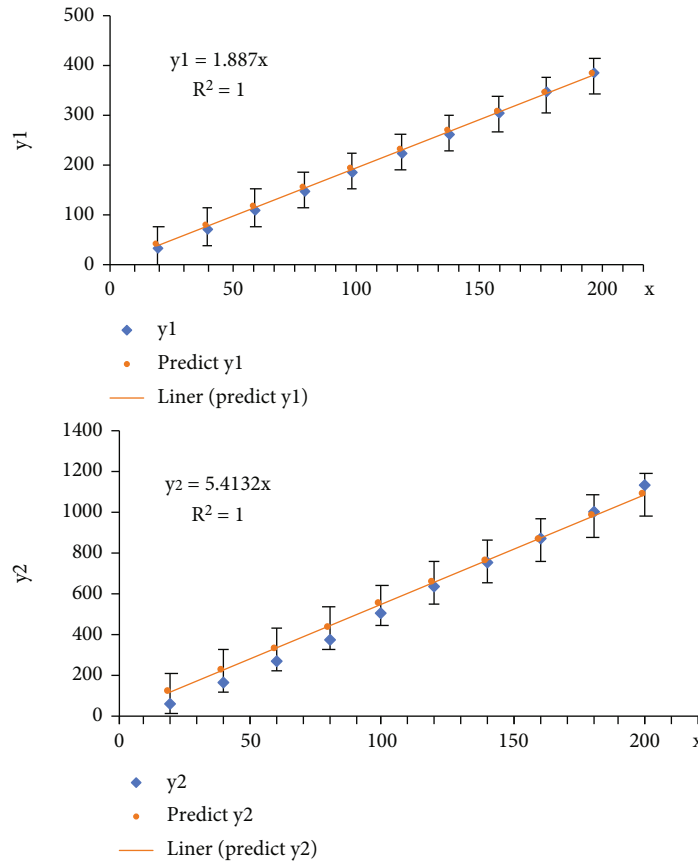
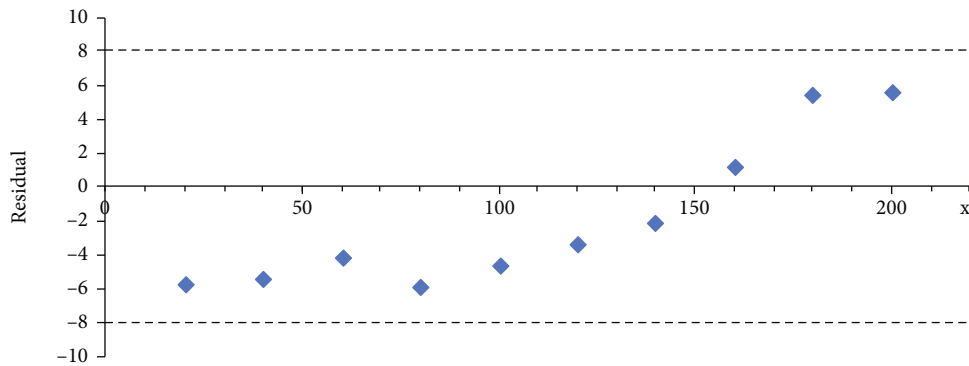FIGURE 23: Linear regression analysis.

FIGURE 24: 2D data residual plot.

or tetrahedrons formed after the Delaunay division and the number of nodes n satisfy a linear relationship.

6.2. A Setting the Distance Constraint Value K between Anchor Nodes. Experiment 2. Setting the distance constraint value k between Delaunay generated anchor nodes.

Data in Tables 1 and 2 were tested. First, the first row of 2D data in Table 1 reveals that when the number of nodes is small, the calculated area accounts for a low percentage. This result likely occurs because the distribution of a small number of randomly generated anchor nodes is uneven, and the distance between nodes is too close, which results in a low

proportion of the area or volume of the Delaunay network. After these analyses, when the number of anchor nodes is small, initial constraints were added to the distance between the nodes generated randomly to increase the proportion of the area divided by the Delaunay method; the minimum constraint distance k is set. The distance $d(s_i, s_j) \leq k$ between each node Delaunay is constrained to make it evenly distributed, where k is a fixed value that can be obtained through multiple experiments.

In experiment 2, we set the phase parameters as $L = 200 \times 200$, $n = 30$, and $k = 25$ for verification. As listed in Table 3, the experiment randomly generated 10 sets of 2D
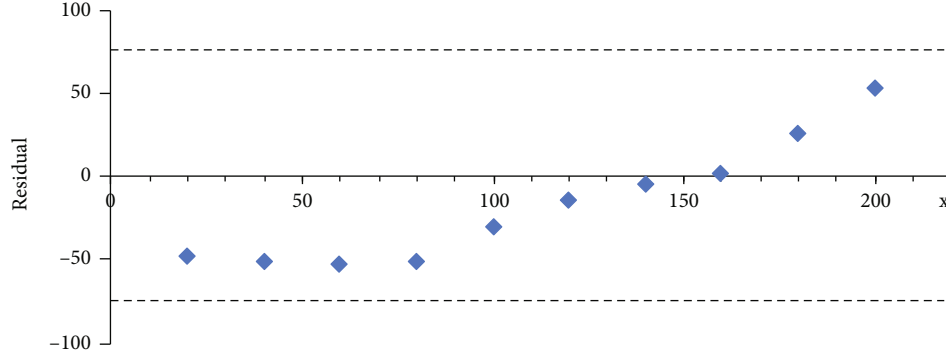
FIGURE 25: 3D data residual plot.

TABLE 3: Setting condition k, the data of the 2D Delaunay graph.

Serial number	Delaunay area S_0 (m^2)	Total area ratio S_0/S_L (%)
1	3.3977e+04	84.94%
2	3.3350e+04	83.38%
3	3.2263e+04	80.66%
4	3.3672e+04	84.18%
5	3.4795e+04	86.99%
6	3.4602e+04	86.51%
7	3.0426e+04	76.04%
8	3.2313e+04	80.78%
9	3.3888e+04	84.72%
10	3.5203e+04	88.01%

TABLE 4: Setting condition k, the data of the 3D Delaunay graph.

Serial number	Delaunay volume V (m^3)	Total volume ratio V/V_0 (%)
1	3.3977e+04	84.94%
2	3.3350e+04	83.38%
3	3.2263e+04	80.66%
4	3.3672e+04	84.18%
5	3.4795e+04	86.99%
6	3.4602e+04	86.51%
7	3.0426e+04	76.04%
8	3.2313e+04	80.78%
9	3.3888e+04	84.72%
10	3.5203e+04	88.01%

data and calculated the total proportion S_0/S_L of each group of Delaunay generated areas S_o and total areas S_L. By comparing the data in Table 3 with the data in the first row of Table 1, setting the constraint distance value k between anchor nodes can increase the area ratio of the anchor nodes. By setting k, the utilization rate of the anchor node can be improved, and the area of the divided area can be increased. Second, the calculated average of the total area ratio of the 10 sets of data in Table 4 is 83.62%, which is an average increase of 10.18% compared with 73.44% in Table 1.

Similarly, Table 4 lists the test results of the Delaunay method, which randomly generates 3D data to verify the influence of the distance k between the constraint nodes on the Delaunay volume. First, we set the relevant parameters $L = 200 \times 200 \times 200$, $n = 70$, and $k = 45$ for the experiments. In the experiment, the 3D volume data V formed by the anchor nodes divided by Delaunay and 10 sets of data for the total volume of V_0/V_L were obtained (as listed in Table 4).

The calculated average volume ratio of Table 4 is 80.17%, which is an average increase of 19.26% compared with the 60.91% data value in Table 2 under the same conditions. Therefore, the verification and analysis of the above two sets of data reveal that when the data were initially randomly generated; adding the constraint value k of the distance between anchor nodes could increase the Delaunay generation area or volume. Consequently, the area and volume ratio of Delaunay's divided areas increased, rendering the layout of the anchor nodes reasonable.

7. Comparison Analysis

In this section, the proposed 2D-DPTL algorithm was compared with the centroid, amorphous, and APIT algorithms [29]. The degree of irregularity (DOI) represents the degree of irregularity of the signal and generally is a value in the range of [0,1]. DOI is defined as the distance change of the largest wireless signal per unit degree change in the signal propagation direction [29]. When the DOI value is zero, this indicates that the signal model is an ideal circular signal model. In this experiment, the node was deployed in an area L with a range of 200×200, and the communication range of the node was set to $R_c = 20$ m. Through experimental tests, we set the relevant parameters of the RSSI path loss model as follows: noise standard deviation $\sigma_k = 4$, path loss value $\eta_k = 2.5$, and $P_k = -24$. Next, the neighbour density (ND), anchor percentage deployment method (random deployment or uniform deployment), and other parameters were selected to verify the accuracy of the algorithm.

7.1. *Effect of Changing the Number of Nodes on Average Localization Error.* Experiment 3. Effect of changing the number of anchor nodes on the average localization error (ALE) of the algorithm.
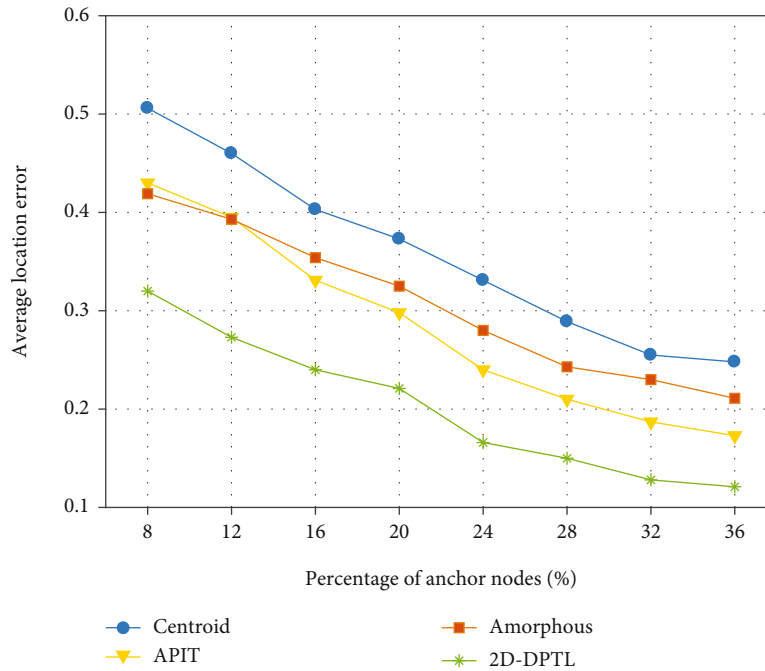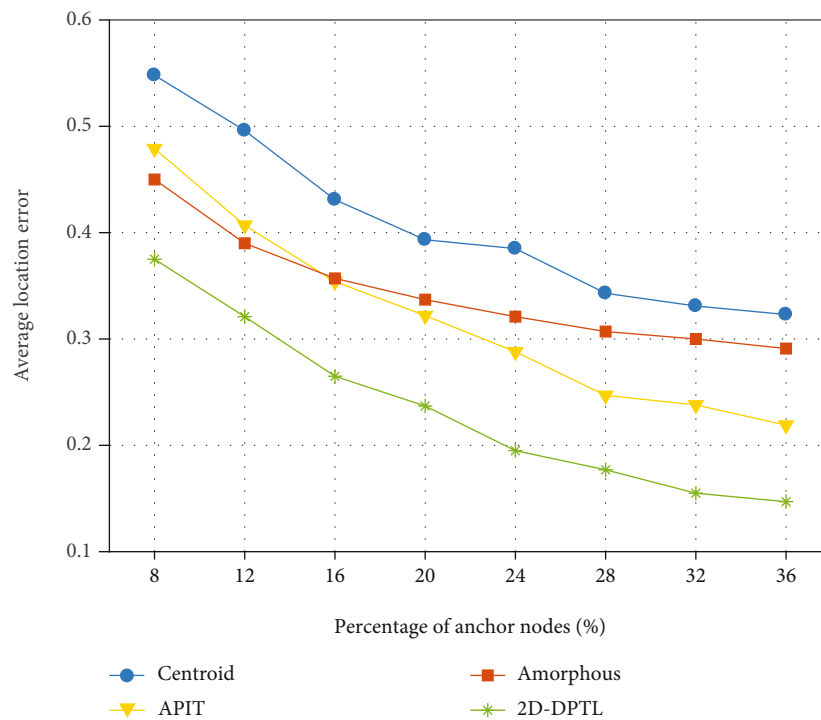
FIGURE 26: Uniform node deployment.

FIGURE 27: Random node deployment.

First, we analysed the effect of the change in the number of anchor nodes on the ALE and compared the performance of the proposed algorithm with other algorithms. Next, the parameters DOI = 0 and ND = 8 were set, and two sets of experiments were designed with various node deployment methods. As displayed in Figure 26, when nodes were deployed uniformly, the ALE of the five algorithms decreased as the percentage of anchor nodes increased. However, the ALE values of the various algorithms differed considerably. The centroid algorithm (CA) exhibited the highest ALE and did not consider the optimization of the anchor node layout, and the calculation method is not rigorous. The proposed 2D-DPTL algorithm exhibited a lower ALE value than other algorithms because the distance
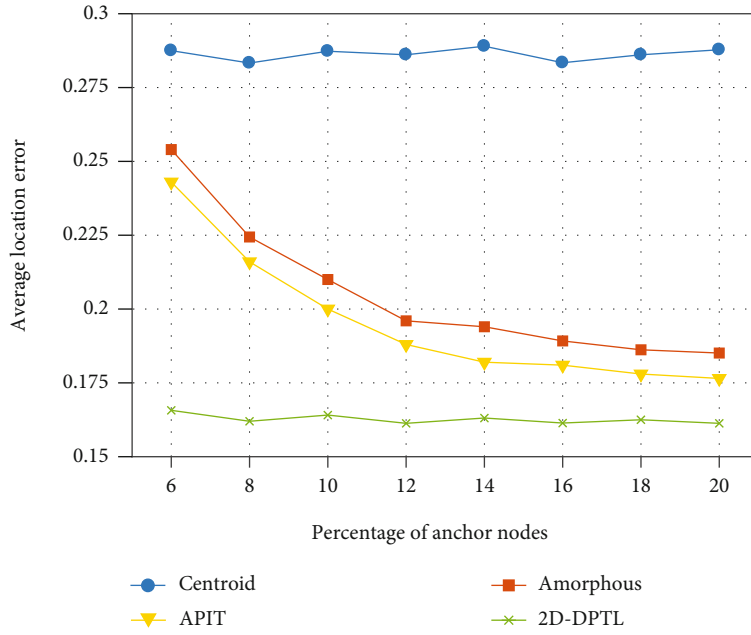
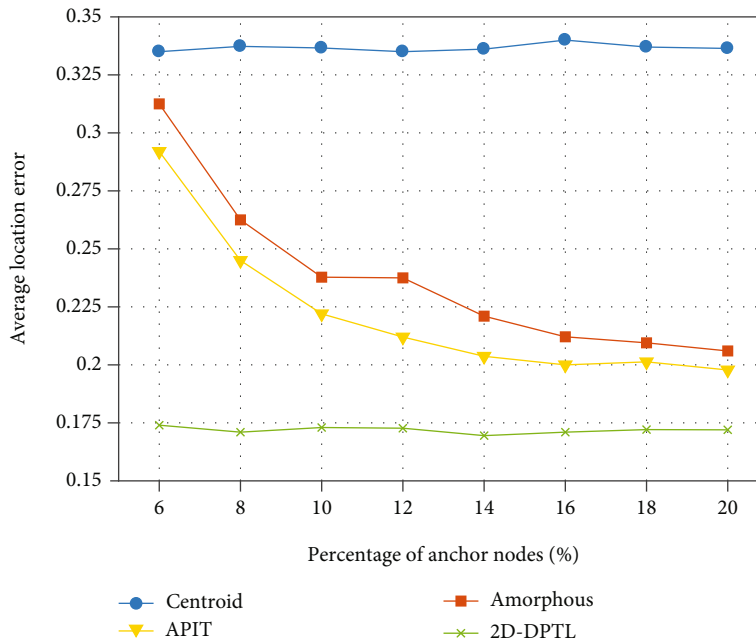FIGURE 28: ALE of all algorithms when DOI = 0.

FIGURE 29: ALE of all algorithms when DOI = 0.1.

judgement condition between nodes was added to optimize the node layout. Consequently, the proposed algorithm divided the target area twice to improve the localization accuracy and reduce the ALE of the algorithm.

Comparing Figures 26 and 27 reveals that the deployment methods differed considerably, and under the same values of other indicators, the ALE value of randomly deployed nodes was greater than that of uniformly deployed nodes. This result likely occurred because the uniform deployment of anchor nodes rendered the layout of nodes

reasonable and increased the accuracy of target localization. Consequently, the ALE of the proposed algorithm was higher than that of other algorithms because the Delaunay method could optimally divide node's location according to its location coordinates from an optimal triangle network, and the location of the target was estimated, which improved location accuracy. Second, the proposed algorithm performed a secondary calculation on the problem of target's excessive localization range and reduced the ALE value of the overall network.

7.2. Effect of the Density of Neighbour Nodes on ALE. Experiment 4. Effect of changing the density of neighbouring nodes on the ALE when the DOI differed.

The effect of the change in the neighbour node density of each algorithm on its ALE was verified. With $AP = 28$ and uniform deployment of nodes, we set DOI = 0 and DOI = 0.1 to conduct two sets of experiments, as displayed in Figures 28 and 29, which reveal that the CA exhibited the highest ALE because this algorithm estimated the centre of mass coordinates as the target localisation and failed to reduce the localisation error. The proposed algorithm used a phased calculation method to reduce ALE.

In Figure 28, the ALE values of the amorphous and APIT algorithms decreased as the density of neighbouring nodes increased, whereas the ALEs of the CA and the proposed algorithm were not affected by a change in neighbouring node density. These results were attributed to the absence of the interaction between the target node and neighbouring nodes. Figures 28 and 29 reveal that the improvement in signal irregularity reduces the average localisation accuracy of the algorithm. Consequently, these experiments reveal that the localisation accuracy of the proposed algorithm was superior to that of the other algorithms and can be maintained within a certain range.

8. Conclusions

The Delaunay division method can optimize the layout of 2D nodes and can be applied to the network layout of 3D sensor nodes based on the location coordinates of the nodes. Delaunay triangle division can improve the flexibility of the node layout and strengthen the correlation between the anchor nodes. Therefore, the Delaunay partition method was used to optimize the Delaunay network layout of randomly deployed anchor nodes. Two localization algorithms were designed to accurately estimate the position of the target in 2D and 3D coordinate systems. The proposed algorithm has limitations in the network life cycle and delay and must be verified in a real-world scenario. The results of this study provide a critical important direction for future research.

Data Availability

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

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

References

[1] S. Tomic, M. Beko, and R. Dinis, "3-D target localization in wireless sensor networks using RSS and AoA measurements," *IEEE Transactions on Vehicular Technology*, vol. 66, no. 4, pp. 3197–3210, 2017.

[2] P. Giri, K. Ng, and W. Phillips, "Wireless sensor network system for landslide monitoring and warning," *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 4, pp. 1210–1220, 2018.

[3] Z. Iqbal, K. Kim, and H. N. Lee, "A cooperative wireless sensor network for indoor industrial monitoring," *IEEE Transactions on Industrial Informatics*, vol. 13, no. 2, pp. 482–491, 2017.

[4] A. M. Badescu and L. Cotofana, "A wireless sensor network to monitor and protect tigers in the wild," *Ecological Indicators*, vol. 57, pp. 447–451, 2015.

[5] S. P. Singh and S. C. Sharma, "Range free localization techniques in wireless sensor networks: a review," *Procedia Computer Science*, vol. 57, pp. 7–16, 2015.

[6] H. P. Mistry and N. H. Mistry, "RSSI based localization scheme in wireless sensor networks: a survey," in *2015 Fifth International Conference on Advanced Computing & Communication Technologies*, pp. 647–652, Haryana, India, February 2015.

[7] L. Zhang, Z. Yang, S. Zhang, and H. Yang, "Three-dimensional localization algorithm of WSN nodes based on RSSI-TOA and single mobile anchor node," *Journal of Electrical and Computer Engineering*, vol. 2019, Article ID 4043106, 8 pages, 2019.

[8] S. I. Lopes, J. M. Vieira, J. Reis, D. Albuquerque, and N. B. Carvalho, "Accurate smartphone indoor positioning using a WSN infrastructure and non-invasive audio for TDoA estimation," *Pervasive and Mobile Computing*, vol. 20, pp. 29–46, 2015.

[9] S. Tomic, M. Beko, and R. Dinis, "Distributed RSS-AoA based localization with unknown transmit powers," *IEEE Wireless Communications Letters*, vol. 5, no. 4, pp. 392–395, 2016.

[10] S. Chang, Y. Li, X. Yang, H. Wang, W. Hu, and Y. Wu, "A novel localization method based on RSS-AOA combined measurements by using polarized identity," *IEEE Sensors Journal*, vol. 19, no. 4, pp. 1463–1470, 2018.

[11] Y. Yao, Q. Han, X. Xu, and N. Jiang, "A RSSI-based distributed weighted search localization algorithm for WSNs," *International Journal of Distributed Sensor Networks*, vol. 11, no. 4, Article ID 293403, 2015.

[12] H. Ahmadi and R. Bouallegue, "Exploiting machine learning strategies and RSSI for localization in wireless sensor networks: a survey," in *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1150–1154, Valencia, Spain, June 2017.

[13] H. Ahmadi, F. Viani, A. Polo, and R. Bouallegue, "An improved anchor selection strategy for wireless localization of WSN nodes," in *2016 IEEE Symposium on Computers and Communication*, pp. 108–113, Messina, Italy, 2016.

[14] W. Wang, X. Liu, M. Li, Z. Wang, and C. Wang, "Optimizing node localization in wireless sensor networks based on received signal strength indicator," *IEEE Access*, vol. 7, pp. 73880–73889, 2019.

[15] B. Shang, J. Tan, X. Hong et al., "Spatiotemporal radio tomographic imaging with Bayesian compressive sensing for RSS-based indoor target localization," in *Cloud Computing and Security. ICCCS 2017. Lecture Notes in Computer Science, vol 10603*, X. Sun, H. C. Chao, X. You, and E. Bertino, Eds., pp. 528–540, Springer, Cham, 2017.

[16] I. Ullah, Y. Liu, X. Su, and P. Kim, "Efficient and accurate target localization in underwater environment," *IEEE Access*, vol. 7, pp. 101415–101426, 2019.

[17] K. N. Sudhakar and D. K. Anvekar, "Improved RSSI localization technique for mobile WSN using genetic operators," in *2018 3rd IEEE International Conference on Recent Trends in Electronics*, pp. 617–622, Bangalore, India, May 2018.

[18] A. Kargar-Barzi and A. Mahani, "H–V scan and diagonal trajectory: accurate and low power localization algorithms in WSNs," *Journal of Ambient Intelligence and Humanized Computing*, vol. 11, pp. 2871–2882, 2019.

[19] Y. Bae, "Robust localization for robot and IoT using RSSI," *Energies*, vol. 12, p. 2212, 2019.

[20] A. Ademuwagun, "RSS-distance rationalization procedure for localization in an indoor environment," *Wireless Sensor Network*, vol. 11, no. 2, pp. 13–33, 2019.

[21] Z. Z. Yu and G. Z. Guo, "Improvement of positioning technology based on RSSI in ZigBee networks," *Wireless Personal Communications*, vol. 95, no. 3, pp. 1943–1962, 2017.

[22] H. Zhou, M. Jin, and H. Wu, "A distributed Delaunay triangulation algorithm based on centroidal Voronoi tessellation for wireless sensor networks," in *Proceedings of the fourteenth ACM international symposium on Mobile ad hoc networking and computing*, pp. 59–68, New York, NY, USA, July 2013.

[23] Q. Q. Li, P. Nevalainen, J. Peña Queralta, J. Heikkonen, and T. Westerlund, "Localization in unstructured environments: towards autonomous robots in forests with Delaunay triangulation," *Remote Sensing*, vol. 12, no. 11, p. 1870, 2020.

[24] T. Ahmad, X. J. Li, and B. C. Seet, "Noise reduction scheme for parametric loop division 3D wireless localization algorithm based on extended Kalman filtering," *Journal of Sensor and Actuator Networks*, vol. 8, p. 24, 2019.

[25] N. Bulusu, J. Heidemann, and D. Estri, "GPS-less low-cost outdoor localization for very small devices," *IEEE Personal Communications*, vol. 7, no. 5, pp. 28–34, 2000.

[26] X. Tang, L. Tan, A. Hussain, and M. Wang, "Three-dimensional Voronoi diagram–based self-deployment algorithm in IoT sensor networks," *Annals of Telecommunications*, vol. 74, pp. 517–529, 2019.

[27] H. Chizari, M. Hosseini, T. Poston, S. A. Razak, and A. H. Abdullah, "Delaunay triangulation as a new coverage measurement method in wireless sensor network," *Sensors*, vol. 11, no. 3, pp. 3163–3176, 2011.

[28] X. Wang, M. Fu, and H. Zhang, "Target tracking in wireless sensor networks based on the combination of KF and MLE using distance measurements," *IEEE Transactions on Mobile Computing*, vol. 11, pp. 567–576, 2011.

[29] J. Liu, Z. Wang, M. Yao, and Z. Qiu, "VN-APIT: virtual nodes-based range-free APIT localization scheme for WSN," *Wireless Networks*, vol. 22, no. 3, pp. 867–878, 2016.

Research Article

Speed Sensor-Less Control System of Surface-Mounted Permanent Magnet Synchronous Motor Based on Adaptive Feedback Gain Supertwisting Sliding Mode Observer

Lei Zhang , **Jing Bai** , **and Jing Wu**

College of Electrical and Information Engineering Beihua University, Ji Lin 132013, China

Correspondence should be addressed to Jing Bai; jlbyj@163.com

Aiming at the problems of severe chattering and difficulty in low-speed operation of the surface-mounted permanent magnet synchronous motor (SPMSM) sensor-less speed control system based on the traditional sliding mode observer (SMO), this paper proposes a sensor-less control strategy of supertwisting sliding mode observer based on adaptive feedback gain (AFG-STA-SMO). This strategy combines the supertwisting algorithm (STA) with the equivalent feedback principle and designs an adaptive law to compensate for the rotor position error by adjusting the feedback gain coefficient online. Secondly, considering the ripple component in the back electromotive force (back-EMF), the Kalman filter gets a smoother back-EMF signal, further improving the rotor position estimation accuracy. The stability of the system is proved by using the Lyapunov function. Finally, the feasibility and effectiveness of the proposed control strategy are verified by MATLAB/Simulink simulation.

1. Introduction

Surface-mounted permanent magnet synchronous motor has a small size, high power density, high efficiency, low rotor loss, and solid environmental adaptability [1–4]. It is widely used in the wind power generation system and industrial transmission field. In order to realize high-performance control of permanent magnet synchronous motor, it is necessary to install speed and position sensors such as a photoelectric encoder to get accurate rotor position and speed information. However, installing a photoelectric encoder will increase the system's cost and require a high operating environment [5]. As a result, various sensor-less techniques have been developed to estimate rotor position and speed over the past few decades.

At present, speed sensor-less control algorithms are mainly divided into two categories: the high-frequency signal injection method, which depends on the motor's salient spatial characteristics. Accurate rotor position information can be obtained at all speeds, including zero speed, but it is only applicable to the built-in motor with much noise. The second type uses the motor speed parameters (such as the

back-EMF) in the motor fundamental excitation mathematical model to estimate the rotor position and speed, which has significant dynamic performance but is only suitable for the operation range of medium and high-speed. The commonly used fundamental back-EMF observation algorithms include SMO [6–9], model reference adaptive [10–11], and extended Kalman filter [12]. SMO has been widely studied because of its simple implementation, insensitivity to parameter changes, and external interference [13]. However, the chattering caused by the sign switching function is an inevitable problem in SMO. Therefore, the low-pass filter (LPF) was introduced to filter the back-EMF. Since the traditional LPF cut-off frequency is fixed, the back-EMF's ripple component cannot be eliminated, seriously affecting the rotor position's estimation accuracy [14]. In literature [15], an adaptive filter is proposed to filter out the sliding mode noise, but the causes of the system's phase delay to a certain extent. The literature [16–18] proposed an SMO based on the supertwisting algorithm to observe back-EMF to solve this problem. This method can effectively suppress the chattering phenomenon caused by the switching function, and a good control effect can be

obtained in the medium and high-speed range. However, with the decrease of motor speed, the back-EMF gradually decreases, and the observation accuracy also decreases. In order to improve the rotor estimation when the motor runs at low speed, an SMO is based on equivalent feedback proposed in the literature [19]. The observer can feed back on the estimated back-EMF to the stator current observation and calculation to achieve the motor's low-speed operation. However, the feedback gain coefficient is only selected according to the speed, so the algorithm lacks adaptability. Based on the equivalent feedback sliding mode observer, a feedback gain adaptive algorithm is proposed in the literature [20, 21] to realize the rotor angle compensation at different speeds. However, this method still needs to introduce an LPF and compensate for the rotor position delay, which increases the complexity of the system.

In this paper, an STA-SMO with adaptive feedback gain was proposed by combining the equivalent feedback principle with the supertwisting algorithm. The observer feeds back on the estimated back-EMF to the stator current observation and calculation and designs an adaptive law to realize rotor angle compensation at different speeds, which improves the observer's low-speed performance and simplifies the selection of sliding mode gain. Simultaneously, the Kalman filter is used to filter out the ripple component in the back-EMF and further improve the sensor-less control precision; the Lyapunov function analyzes the system's stability.

2. Design of SMO

2.1. Mathematical Model of SPMSM.
The mathematical model of surface-mounted PMSM, the α-β stationary coordinate system, is [22]

$$\begin{bmatrix} U\alpha \\ U\beta \end{bmatrix} = \begin{bmatrix} R + \dfrac{d}{dt}L & 0 \\ 0 & R + \dfrac{d}{dt}L \end{bmatrix} \begin{bmatrix} i\alpha \\ i\beta \end{bmatrix} + \begin{bmatrix} E\alpha \\ E\beta \end{bmatrix}, \qquad (1)$$

where $U\alpha\beta$, $i\alpha\beta$, $E\alpha\beta$, R, and L are α-β axis voltages, α-β axis currents, α-β axis back-EMFs, stator resistance, and stator inductance, respectively. And $E\alpha\beta$ satisfies

$$\begin{bmatrix} E\alpha \\ E\beta \end{bmatrix} = (\omega e \psi f) \begin{bmatrix} -\sin\theta e \\ \cos\theta e \end{bmatrix}, \qquad (2)$$

where ψf is the flux linkage, θe is the rotor position angle, and ωe is the rotational speed.

The derivative of equation (2) is obtained:

$$\frac{d}{dt}\begin{bmatrix} E\alpha \\ E\beta \end{bmatrix} = \omega e \begin{bmatrix} -\psi f \omega e \cos\theta e \\ -\psi f \omega e \sin\theta e \end{bmatrix} = \omega e \begin{bmatrix} -E\beta \\ E\alpha \end{bmatrix}. \qquad (3)$$

As shown in equation (3), the back-EMF contains the rotor position and speed information.

2.2. Traditional SMO.
In order to obtain the back-EMF, equation (1) is rewritten as follows:

$$\frac{d}{dt}\begin{bmatrix} i\alpha \\ i\beta \end{bmatrix} = -\frac{R}{L}\begin{bmatrix} i\alpha \\ i\beta \end{bmatrix} + \frac{1}{L}\begin{bmatrix} U\alpha \\ U\beta \end{bmatrix} - \frac{1}{L}\begin{bmatrix} E\alpha \\ E\beta \end{bmatrix}. \qquad (4)$$

According to equation (4), the traditional SMO can be designed as

$$\frac{d}{dt}\begin{bmatrix} \hat{i}\alpha \\ \hat{i}\beta \end{bmatrix} = -\frac{R}{L}\begin{bmatrix} \hat{i}\alpha \\ \hat{i}\beta \end{bmatrix} + \frac{1}{L}\begin{bmatrix} U\alpha \\ U\beta \end{bmatrix} - \frac{1}{Ls}\begin{bmatrix} v\alpha \\ v\beta \end{bmatrix}, \qquad (5)$$

where $\hat{i}\alpha\beta$ is the observed value of stator current, $U\alpha\beta$ is the control input of the observer, and $v\alpha\beta$ is the estimated value of the back electromotive force.

$$\begin{bmatrix} v\alpha \\ v\beta \end{bmatrix} = \begin{bmatrix} k\,\mathrm{sign}\,(\hat{i}\alpha - i\alpha) \\ k\,\mathrm{sign}\,(\hat{i}\beta - i\beta) \end{bmatrix}, \qquad (6)$$

where sign () is the signum function and k is the sliding mode gain coefficient.

From equation (6), it can be seen that there is a discontinuous function sign () in the estimated back-EMF value, which makes the back-EMF contain a large number of sliding mode noise and harmonic components, so it needs to introduce a LPF to filter it. Since the phase and amplitude errors are introduced into the LPF, it must be compensated, which increases the complexity of the system.

In order to further improve the low-speed performance of speed sensor-less control, a SMO based on equivalent feedback electromotive force was proposed in literature [19], as shown in the following equation:

$$\frac{d}{dt}\begin{bmatrix} \hat{i}\alpha \\ \hat{i}\beta \end{bmatrix} = -\frac{R}{L}\begin{bmatrix} \hat{i}\alpha \\ \hat{i}\beta \end{bmatrix} + \frac{1}{L}\begin{bmatrix} U\alpha \\ U\beta \end{bmatrix} - \frac{1}{Ls}\begin{bmatrix} v\alpha + \varepsilon v\alpha\mathrm{eq} \\ v\beta + \varepsilon v\beta\mathrm{eq} \end{bmatrix}, \qquad (7)$$

where ε is the feedback gain coefficient and $v\alpha\beta$eq are the equivalent feedback electromotive force values.

$$\begin{bmatrix} v\alpha\mathrm{eq} \\ v\beta\mathrm{eq} \end{bmatrix} = \begin{bmatrix} \dfrac{\omega c}{\tau s + \omega c}v\alpha \\ \dfrac{\omega c}{\tau s + \omega c}v\beta \end{bmatrix}, \qquad (8)$$

where ωc is the cut-off frequency of the LPF and τ is the time constant of the LPF.

When the motor is running at low speed, the SMO based on the equivalent feedback EMF can improve the value of the equivalent feedback EMF by reasonably designing parameter ε, so as to solve the problem that the response EMF is too small and the rotor position cannot be estimated effectively, and improve the control precision of the system. However, the method still needs to introduce a low-pass filter and conduct behavior compensation.
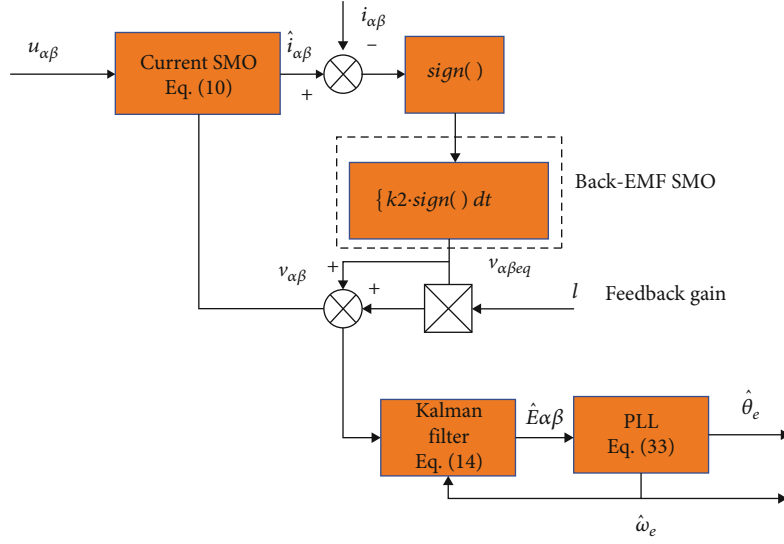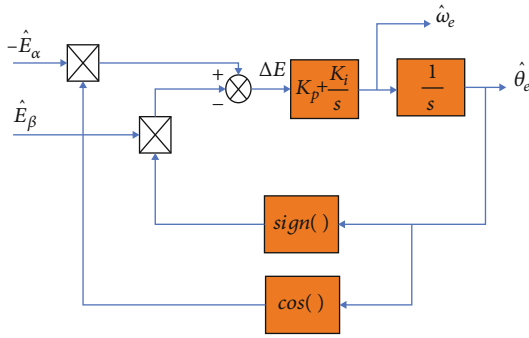
FIGURE 1: Block diagram of AFG-STA-SMO.

FIGURE 2: Schematic diagrams of PLL.

In order to retain the advantages of the equivalent feedback SMO and avoid using a LPF, a new sliding mode speed sensor-less control method was designed in literature [23]. In this method, a new state observer is designed to observe the equivalent feedback signal $v\alpha\beta$eq, thus avoiding the use of a LPF. However, the design and debugging of the state observer are complicated.

3. Design of AFG-STA-SMO

3.1. Supertwisting Algorithm.
The general form of the supertwisting algorithm [17] is as follows:

$$\frac{d\hat{x}1}{dt} = -k1|\tilde{x}1|^{1/2} \, \text{sign} \, (\tilde{x}1) + x2 + \rho_1,$$
$$\frac{d\hat{x}2}{dt} = -k2 \, \text{sign} \, (\tilde{x}1) + \rho_2, \tag{9}$$

where xi, $\hat{x}i$, $\tilde{x}i$, ki, ρi, and sign () are state variables, estimation of state variables, the error between estimated and actual state variables, sliding mode gain coefficient, perturbation term, and signum function, respectively.
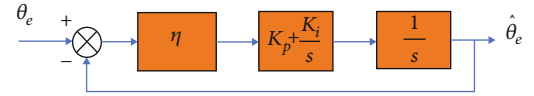

FIGURE 3: Equivalent schematic diagrams of PLL.

3.2. AFG-STA-SMO.
Since SPMSM high-precision vector control requires accurate motor speed and rotor position information, traditional SMO has parameter perturbation and chattering. This paper combines the principle of equivalent feedback with STA-SMO, and the design is as follows:

$$\begin{cases} \frac{d}{dt}\begin{bmatrix}\hat{i}\alpha\\\hat{i}\beta\end{bmatrix} = -\frac{R}{L}\begin{bmatrix}\hat{i}\alpha\\\hat{i}\beta\end{bmatrix} + \frac{1}{L}\begin{bmatrix}U\alpha\\U\beta\end{bmatrix} - \frac{1}{L}\begin{bmatrix}v\alpha + l \cdot v\alpha eq\\v\beta + l \cdot v\beta eq\end{bmatrix} - \begin{bmatrix}k1 \cdot |\tilde{i}\alpha|^{1/2} \cdot \text{sign} \, (\tilde{i}\alpha)\\k1 \cdot |\tilde{i}\beta|^{1/2} \cdot \text{sign} \, (\tilde{i}\beta)\end{bmatrix}, \\ \frac{d}{dt}\begin{bmatrix}v\alpha\\v\beta\end{bmatrix} = \begin{bmatrix}k2 \cdot \text{sign} \, (\tilde{i}\alpha)\\k2 \cdot \text{sign} \, (\tilde{i}\beta)\end{bmatrix}, \\ \begin{bmatrix}v\alpha eq\\v\beta eq\end{bmatrix} = \begin{bmatrix}v\alpha\\v\beta\end{bmatrix}, \end{cases} \tag{10}$$

where l is the feedback gain coefficient, $\hat{i}\alpha\beta$ are the observation value of the stator currents, $U\alpha\beta$ are α-β axis voltages, $-((R/L)i\alpha) + ((1/L)U\alpha)$ and $-((R/L)i\beta) + ((1/L)U\beta)$ are regarded as the perturbation terms, $v\alpha = \int k2 \cdot \text{sign} \, (\tilde{i}\alpha)dt$ and $v\beta = \int k2 \cdot \text{sign} \, (\tilde{i}\beta)dt$ are the estimated value of back-EMFs, and $v\alpha$eq and $v\beta$eq are the equivalent feedback electromotive force values.

The difference between equations (4) and (10) can be obtained:

$$\frac{d}{dt}\begin{bmatrix}\tilde{i}\alpha\\\tilde{i}\beta\end{bmatrix} = -\frac{Rs}{Ls}\begin{bmatrix}\tilde{i}\alpha\\\tilde{i}\beta\end{bmatrix} - \frac{1}{Ls}\begin{bmatrix}\tilde{e}\alpha\\\tilde{e}\beta\end{bmatrix} - \begin{bmatrix}k1 \cdot |\tilde{i}\alpha|^{1/2} \cdot \text{sign} \, (\tilde{i}\alpha)\\k1 \cdot |\tilde{i}\beta|^{1/2} \cdot \text{sign} \, (\tilde{i}\beta)\end{bmatrix}, \tag{11}$$
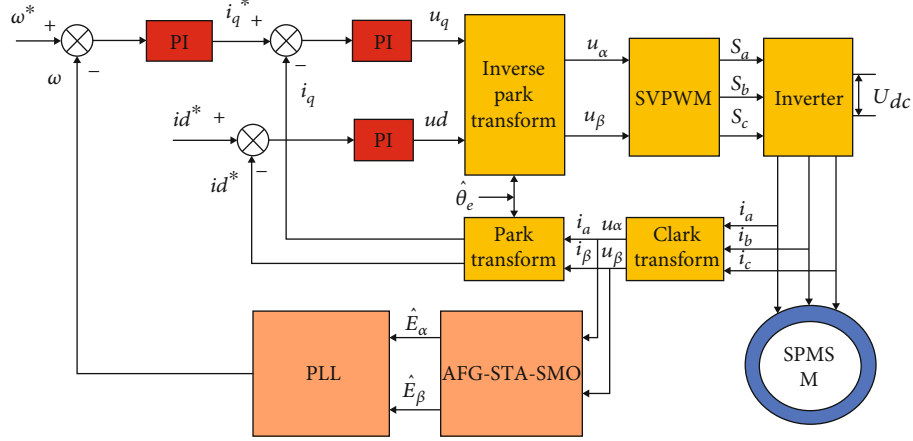
FIGURE 4: SPMSM speed sensor-less control block diagram of AFG-STA-SMO.

where $\tilde{i\alpha} = \hat{i\alpha} - i\alpha$, $\tilde{i\beta} = \hat{i\beta} - i\beta$, $\tilde{e\alpha} = v\alpha + lv\alpha eq - E\alpha$, and $\tilde{e\beta} = v\beta + lv\beta eq - E\beta$.

After the state variable of the observer reaches the sliding mode surface $\tilde{i\alpha} = 0$ and $\tilde{i\beta} = 0$, according to the sliding mode variable structure equivalent principle, we can get

$$\begin{bmatrix} E\alpha \\ E\beta \end{bmatrix} = \begin{bmatrix} v\alpha + lv\alpha eq \\ v\beta + lv\beta eq \end{bmatrix} = \begin{bmatrix} (1+l) \cdot \int k2 \cdot \text{sign}\,(\tilde{i\alpha})dt \\ (1+l) \cdot \int k2 \cdot \text{sign}\,(\tilde{i\beta})dt \end{bmatrix}. \tag{12}$$

From equation (12), the integral term $v\alpha\beta$ filters out high-frequency sliding mode noise, so introducing low-pass filters and phase compensation is avoided, and the system algorithm is simplified. It can be seen from equation (12) that the sliding mode gain $k2$ and feedback gain l are proportional to the speed. When the motor is running at high speed, the sliding-mode gain $k2$ and the feedback gain l are larger. When the motor runs at low speed, the sliding-mode gain $k2$ and the feedback gain l are smaller.

3.3. Optimization of AFG-STA-SMO. The derivative of equation (12) is obtained:

$$\begin{cases} \dfrac{d}{dt}E\alpha = -\omega e \cdot E\beta = (1+l) \cdot k2 \cdot \text{sign}\,(\tilde{i\alpha}), \\ \dfrac{d}{dt}E\beta = \omega e \cdot E\alpha = (1+l) \cdot k2 \cdot \text{sign}\,(\tilde{i\beta}). \end{cases} \tag{13}$$

Although the harmonic component is effectively filtered by integral operation, the back-EMF's observed value still contains a ripple component, which will lead to error if used to estimate the rotor position directly. Therefore, the Kalman filter is used in this paper to obtain a smoother back-EMF signal, which further improves the rotor position estimation accuracy.

TABLE 1: SPMSM parameters.

Parameters	Values
Stator resistance R_s	2.875 Ω
Pole-pair number	4
Magnetic flux ψ_f	0.175 Wb
D-axis induction L_d	8.5e-3H
Q-axis induction L_q	8.5e-3H
Damping factor B	0
DC bus voltage U_{dc}	310 V
Pole-pair number	4
Magnetic flux ψ_f	0.175 Wb
Rotational inertia J	0.001 kg·m^2

According to equation (13), the state equation of the Kalman filter is designed as [24]

$$\begin{bmatrix} \dfrac{d}{dt}\hat{E}\alpha \\ \dfrac{d}{dt}\hat{E}\beta \\ \dfrac{d}{dt}\hat{\omega}e \end{bmatrix} = \begin{bmatrix} -\hat{E}\beta\hat{\omega}e - \tilde{E}\alpha\lambda \\ \hat{E}\alpha\hat{\omega}e - \tilde{E}\beta\lambda \\ \hat{E}\beta\tilde{E}\alpha - \hat{E}\alpha\tilde{E}\beta \end{bmatrix}, \tag{14}$$

where λ is the adaptive gain, $\hat{\omega}e$ is the estimated rotational speed, $\hat{E}\alpha$ and $\hat{E}\beta$ are the final observed values of back-EMF, $\tilde{E}\alpha = \hat{E}\alpha - v\alpha - lv\alpha eq$, and $\tilde{E}\beta = \hat{E}\beta - v\beta - lv\beta eq$.

Considering that the adaptive gain coefficient which is too large will cause the system to be unstable and too small will cause the system to converge too slowly, so this paper designs the adaptive gain according to the actual speed of the motor as
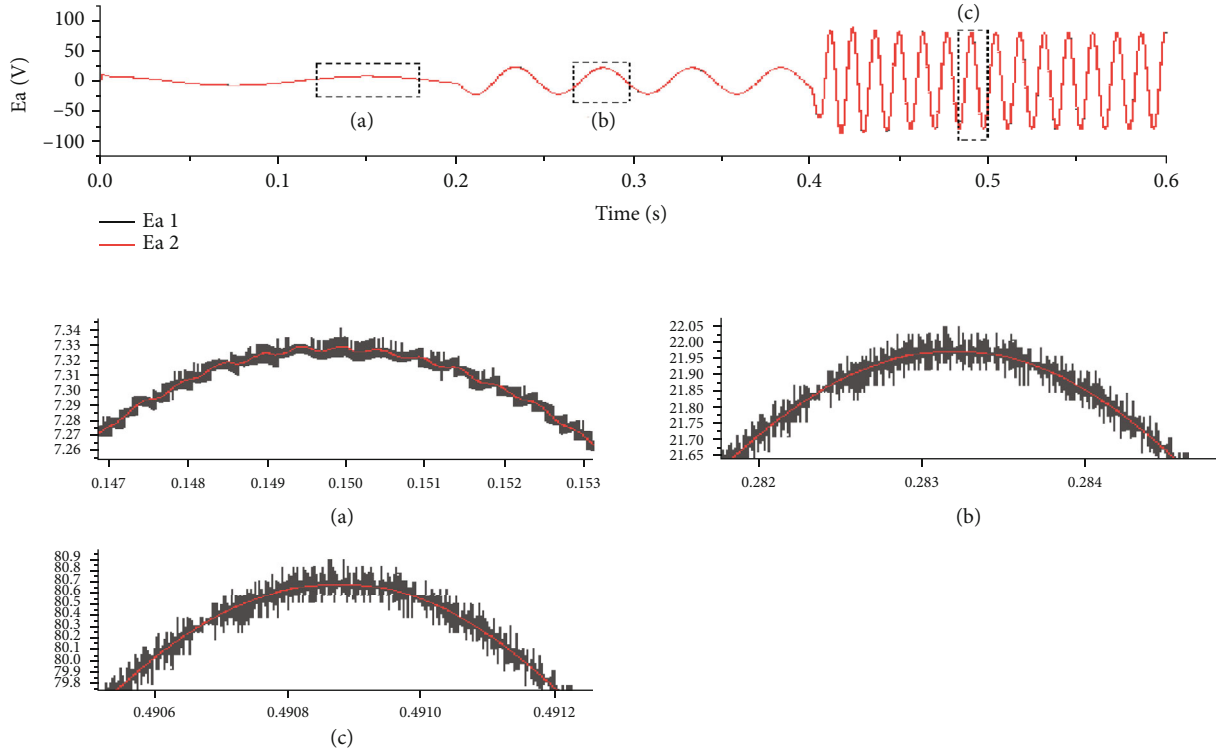
$$\lambda = \lambda a + \kappa\omega e, \tag{15}$$

Figure 5: Back-EMF response diagram.

where λa and κ are both normal numbers and ωe represents the actual speed of the motor.

Stability analysis

$$V = \frac{1}{2}\left(\tilde{E}^2\alpha + \tilde{E}^2\beta + \tilde{\omega}^2 e\right). \quad (16)$$

where $\tilde{\omega}e = \hat{\omega}e - \omega e$.

Since the mechanical time constant is much larger than the electrical time constant, it is considered that the speed does not change within an estimation period. which can be obtained from equation (3) and equation (14):

$$\begin{bmatrix} \dfrac{d}{dt}\tilde{E}\alpha \\ \dfrac{d}{dt}\tilde{E}\beta \\ \dfrac{d}{dt}\tilde{\omega}e \end{bmatrix} = \begin{bmatrix} -\hat{E}\beta\tilde{\omega}e - \tilde{E}\beta\hat{\omega}e - \tilde{E}\alpha\lambda \\ \hat{E}\alpha\tilde{\omega}e + \tilde{E}\alpha\hat{\omega}e - \tilde{E}\beta\lambda \\ \hat{E}\beta\tilde{E}\alpha - \hat{E}\alpha\tilde{E}\beta \end{bmatrix}. \quad (17)$$

Substitute equation (17) into equation (16) and derive

$$\begin{aligned} \frac{d}{dt}V &= \tilde{E}\alpha\frac{d}{dt}\tilde{E}\alpha + \tilde{E}\beta\frac{d}{dt}\tilde{E}\beta + \tilde{\omega}e\frac{d}{dt}\tilde{\omega}e \\ &= -\tilde{E}\alpha\hat{E}\beta\tilde{\omega}e - \tilde{E}\alpha\tilde{E}\beta\hat{\omega}e - \lambda\tilde{E}^2\alpha + \hat{E}\alpha\tilde{E}\beta\tilde{\omega}e + \tilde{E}\alpha\tilde{E}\beta\hat{\omega}e \\ &\quad - \lambda\tilde{E}^2\beta + \tilde{E}\alpha\hat{E}\beta\tilde{\omega}e - \hat{E}\alpha\tilde{E}\beta\tilde{\omega}e = -\lambda\left(\tilde{E}^2\alpha + \tilde{E}^2\beta\right) \le 0. \end{aligned} \quad (18)$$

Table 2: Back-EMF fluctuation.

Result	Ea1	Ea2	Unit
Back-EMF fluctuation (a)	7.34~7.32	7.325~7.315	V
Back-EMF fluctuation (b)	22.05~21.925	22.025~22.015	V
Back-EMF fluctuation (c)	80.9~80.5	80.7~80.69	V

Therefore, according to the Lyapunov stability criterion, the system with the Kalman filter is stable. The structure schematic diagram of AFG-STA-SMO is shown in Figure 1.

3.4. System Stability Analysis. The Lyapunov function can define as

$$V1 = \frac{1}{2}\left(\tilde{i}\alpha^2 + \tilde{i}\beta^2\right). \quad (19)$$

From equations (19) and (11), it can be obtained that

$$\frac{d}{dt}V1 = -\frac{R}{L}\left(\tilde{i}^2\alpha + \tilde{i}^2\beta\right) - \frac{1}{L}\left(\tilde{e}\alpha\tilde{i}\alpha + \tilde{e}\beta\tilde{i}\beta\right) - k1\left(|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2}\right) < 0. \quad (20)$$

In order to ensure the convergence of the current-sliding mode observer, $(d/dt)V1 < 0$ must satisfy; then, the sliding mode gain $k1$ can be obtained as follows:
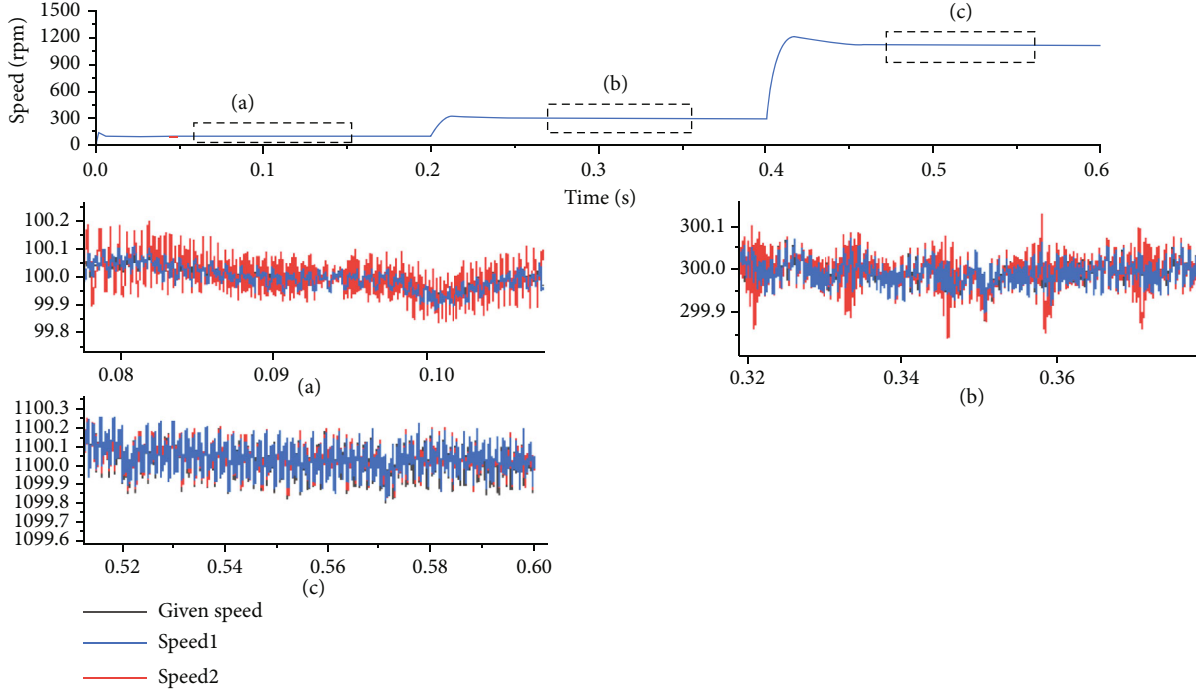
FIGURE 6: Speed response diagram.

$$k1 > \sup \left\{ -\frac{R}{L} \cdot \frac{\left(\tilde{i}^2\alpha + \tilde{i}^2\beta\right)}{|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|3/2} - \frac{1}{L} \cdot \frac{\tilde{e}\alpha\tilde{i}\alpha + \tilde{e}\beta\tilde{i}\beta}{|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2}} \right\}, \tag{21}$$

where $\sup \{\}$ represents the upper bound.

When the current-sliding mode observer converges, $(d/dt)\tilde{i}\alpha = \tilde{i}\alpha = (d/dt)\tilde{i}\beta = \tilde{i}\beta = 0$, from equation (11) that

$$\begin{cases} \operatorname{sign}(\tilde{i}\alpha) = -\dfrac{\tilde{e}\alpha}{k1 \cdot L \cdot |\tilde{i}\alpha|^{1/2}}, \\ \operatorname{sign}(\tilde{i}\beta) = -\dfrac{\tilde{e}\beta}{k1 \cdot L \cdot |\tilde{i}\beta|^{1/2}}. \end{cases} \tag{22}$$

Substitute equation (22) into equation (10) and get

$$\begin{cases} \dfrac{d}{dt}(v\alpha + lv\alpha\mathrm{eq}) = -(1+l)\dfrac{k2 \cdot \tilde{e}\alpha}{k1 \cdot Ls \cdot |\tilde{i}\alpha|^{1/2}}, \\ \dfrac{d}{dt}(v\beta + lv\beta\mathrm{eq}) = -(1+l)\dfrac{k2 \cdot \tilde{e}\beta}{k1 \cdot Ls \cdot |\tilde{i}\beta|^{1/2}}. \end{cases} \tag{23}$$

It can be obtained from equations (23) and (3) that

$$\begin{cases} \dfrac{d}{dt}\tilde{e}\alpha = -(1+l)\dfrac{k2 \cdot \tilde{e}\alpha}{k1 \cdot Ls \cdot |\tilde{i}\alpha|^{1/2}} + \omega e \cdot E\beta, \\ \dfrac{d}{dt}\tilde{e}\beta = -(1+l)\dfrac{k2 \cdot \tilde{e}\beta}{k1 \cdot Ls \cdot |\tilde{i}\beta|^{1/2}} - \omega e \cdot E\alpha. \end{cases} \tag{24}$$

Similarly, the Lyapunov function can be defined as

$$V2 = \frac{1}{2}\left(\tilde{e}\alpha^2 + \tilde{e}\beta^2\right). \tag{25}$$

From equations (25) and (24), the following equation can be obtained:

$$\begin{aligned} \frac{d}{dt}V2 &= -(1+l)k2\left[\frac{\tilde{e}^2\alpha}{k1 \cdot L \cdot |\tilde{i}\alpha|^{1/2}} + \frac{\tilde{e}^2\beta}{k1 \cdot L \cdot |\tilde{i}\beta|^{1/2}}\right] \\ &\quad + \omega e \cdot E\beta \cdot \tilde{e}\alpha - \omega e \cdot E\alpha \cdot \tilde{e}\beta < 0. \end{aligned} \tag{26}$$

To ensure the convergence of the current-sliding mode observer, $(d/dt)V2 < 0$ must satisfy; then, the sliding mode gain $k2$ can be obtained as follows:

$$(1+l)k2 > \sup \left\{ \frac{k1 \cdot Ls \cdot \omega e \cdot (E\beta \cdot \tilde{e}\alpha - E\alpha \cdot \tilde{e}\beta)}{\left(\tilde{e}^2\alpha/|\tilde{i}\alpha|^{1/2}\right) + \left(\tilde{e}^2\beta/|\tilde{i}\beta|^{1/2}\right)} \right\}. \tag{27}$$

Equations (21) and (27) give the stability conditions of the established system.

3.5. Adaptive Law Analysis of Feedback Gain Coefficient. In equation (20), $-R/L(\tilde{i}^2\alpha + \tilde{i}^2\beta)$ and $-k1(|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2})$ is always less than 0. A sufficient condition for the existence of equation (20) is constructed as follows:
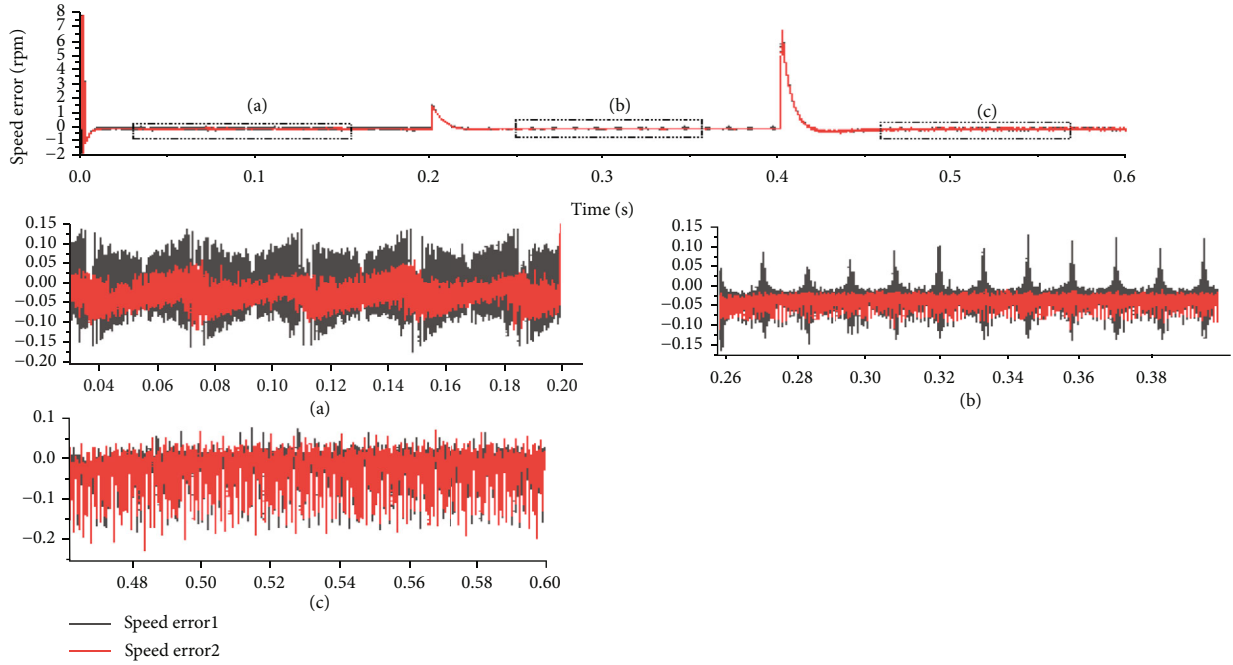
FIGURE 7: Speed error response diagram.

$$\begin{cases} -\dfrac{R}{L}\left(\tilde{i}^2\alpha + \tilde{i}^2\beta\right) - \dfrac{1}{L}\left(\tilde{e}\alpha\tilde{i}\alpha + \tilde{e}\beta\tilde{i}\beta\right) - k1\left(|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2}\right) < 0, & \tilde{i}\alpha > 0, \tilde{i}\beta > 0, \\ -\dfrac{R}{L}\left(\tilde{i}^2\alpha + \tilde{i}^2\beta\right) - \dfrac{1}{L}\left(-\tilde{e}\alpha\tilde{i}\alpha - \tilde{e}\beta\tilde{i}\beta\right) - k1\left(|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2}\right) < 0, & \tilde{i}\alpha < 0, \tilde{i}\beta < 0. \end{cases}$$

$$(28)$$

Since $-R/L(\tilde{i}^2\alpha + \tilde{i}^2\beta) < 0$ and $-k1(|\tilde{i}\alpha|^{3/2} + |\tilde{i}\beta|^{3/2}) < 0$, it can be deduced that the condition satisfying $(d/dt)V1 < 0$ is as follows:

$$(1 + l)k2 > \max\left(|E\alpha|, |E\beta|\right) = |\omega e|\psi f, \tag{29}$$

where ωe is the actual speed and ψf is the flux linkage.

From equation (29), it can be seen that the selection of l affects the size of the sliding mode gain of $k2$. The larger the l is, the lower the limit of the effective range of $k2$ is. Conversely, the smaller the l is, the higher the lower limit of the effective range of $k2$ is.

According to equation (29),

$$l > \frac{|\omega e|\psi f}{k2} - 1, \tag{30}$$

where $k2 > \psi f$.

According to equation (30), $1 > (\psi f/k2) > 0$ is known, so the feedback gain coefficient l can be designed as follows:

$$l = \delta|\omega e| - 1, \tag{31}$$

where δ is a normal number, which is related to the gain of sliding mode $k2$. So $k2$ meet the following conditions:

$$k2 > \frac{\psi f}{\delta}. \tag{32}$$

TABLE 3: Back-EMF fluctuation.

Result	FFG-STA-SMO	AFG-STA-SMO	Unit
Error (a)	0.3225	0.175	Rpm
Error (b)	0.275	0.0975	Rpm
Error (c)	0.27	0.27	Rpm
Position error (a)	0.0043	0.0007	Deg
Position error (b)	0.00037	0.00009	Deg
Position error (c)	0.00016	0.00016	Deg

In order to select δ conveniently, $0 < \delta < 1$ is set in this paper.

4. Rotor Position Estimation

Since the sliding mode control is accompanied by high-frequency chattering in the sliding mode, the high-frequency chattering phenomenon will estimate in the back-EMF. The rotor position estimation method based on the arctangent function will introduce this chattering into the division operation; significantly, when the observed value of back-EMF exceeds zero, the rotor error will amplify. Therefore, PLL [25] is adopted in this paper to replace the arctangent function to extract the rotor's position information, as shown in Figure 2.

Hypothesis $\eta = (Ld - Lq)(\omega eid - piq) + \hat{\omega}e\zeta f$, when $|\theta e - \hat{\theta}e| < \pi/6$ and $\sin(\theta e - \hat{\theta}e) = (\theta e - \hat{\theta}e)$ is established, according to Figure 2, can get the following relationship:

$$\Delta E = -\hat{E}\alpha\cos\hat{\theta}e - \hat{E}\beta\sin\hat{\theta}e = \eta\sin\theta e\cos\hat{\theta}e - \eta\sin\hat{\theta}e\cos\theta e$$
$$= \eta\sin\left(\theta e - \hat{\theta}e\right) \approx \eta\left(\theta e - \hat{\theta}e\right).$$
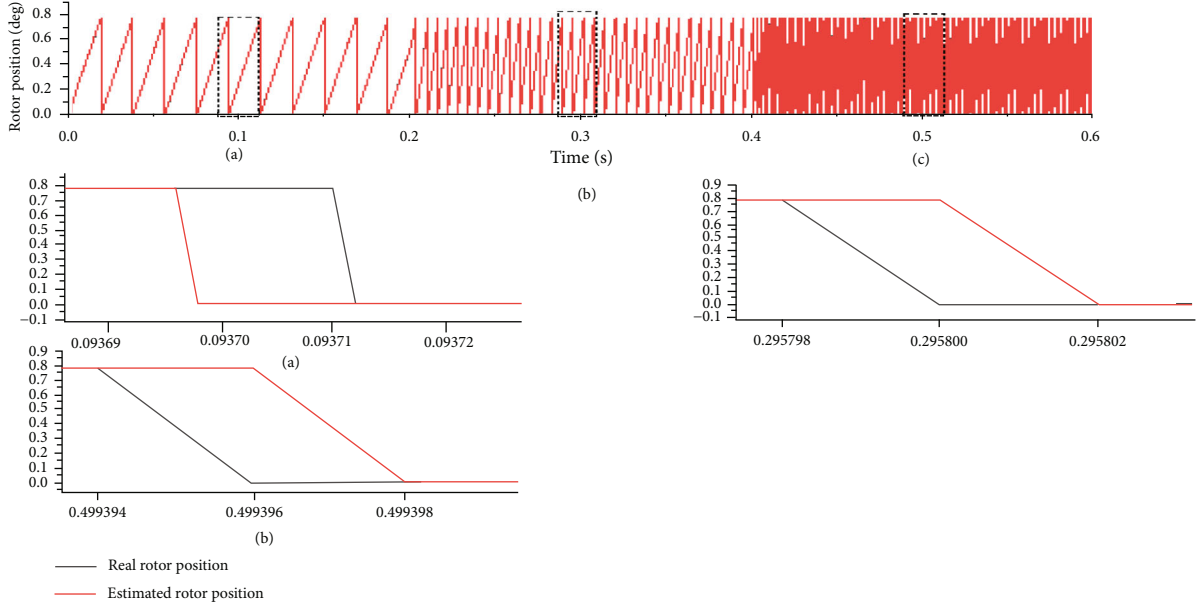
$$(33)$$

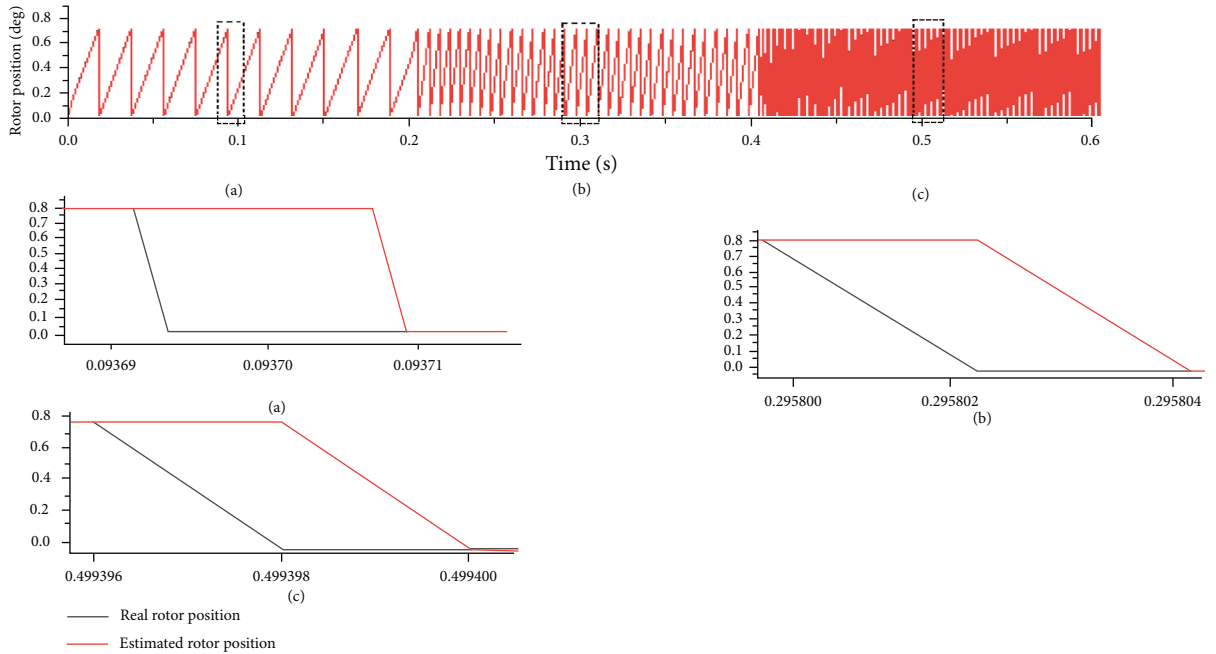Figure 8: Rotor position change with fixed feedback gain.

Figure 9: Rotor position change with adaptive feedback gain.

At this time, the equivalent block diagram of Figure 2 is shown in Figure 3.

According to Figure 3, the transfer function of PLL can be obtained, namely,

$$G(s) = \frac{\widehat{\theta}e}{\theta e} = \frac{2\xi\omega n + \omega^2 n}{s^2 + 2\xi\omega ns + \omega^2 n}, \tag{34}$$

where $\omega n = (Kp/2)\sqrt{\eta/Ki}$, $\xi = \sqrt{\eta Ki}$, and ωn determine the bandwidth of the PI regulator.

5. Simulation

According to Figure 4, MATLAB is used to build a simulation model, and the control scheme adopts vector control based on id = 0. The motor parameters in the simulation experiment are shown in Table 1.

The control system parameters of SPMSM based on the traditional sliding mode observer are as follows: k = 200. The control system parameters of SPMSM based on traditional STA-SMO are as follows: $k_1 = 5000$ and $k_2 = 200000$. The control system parameters of STA-SMO PMSM based on
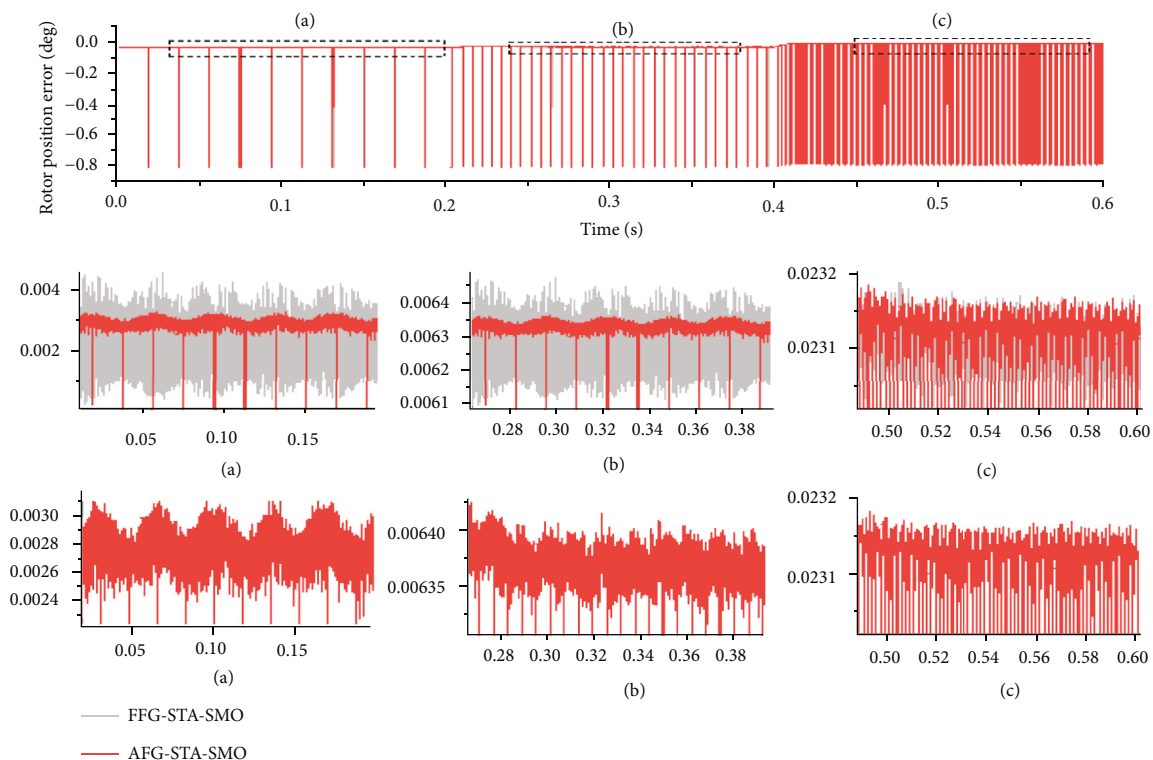
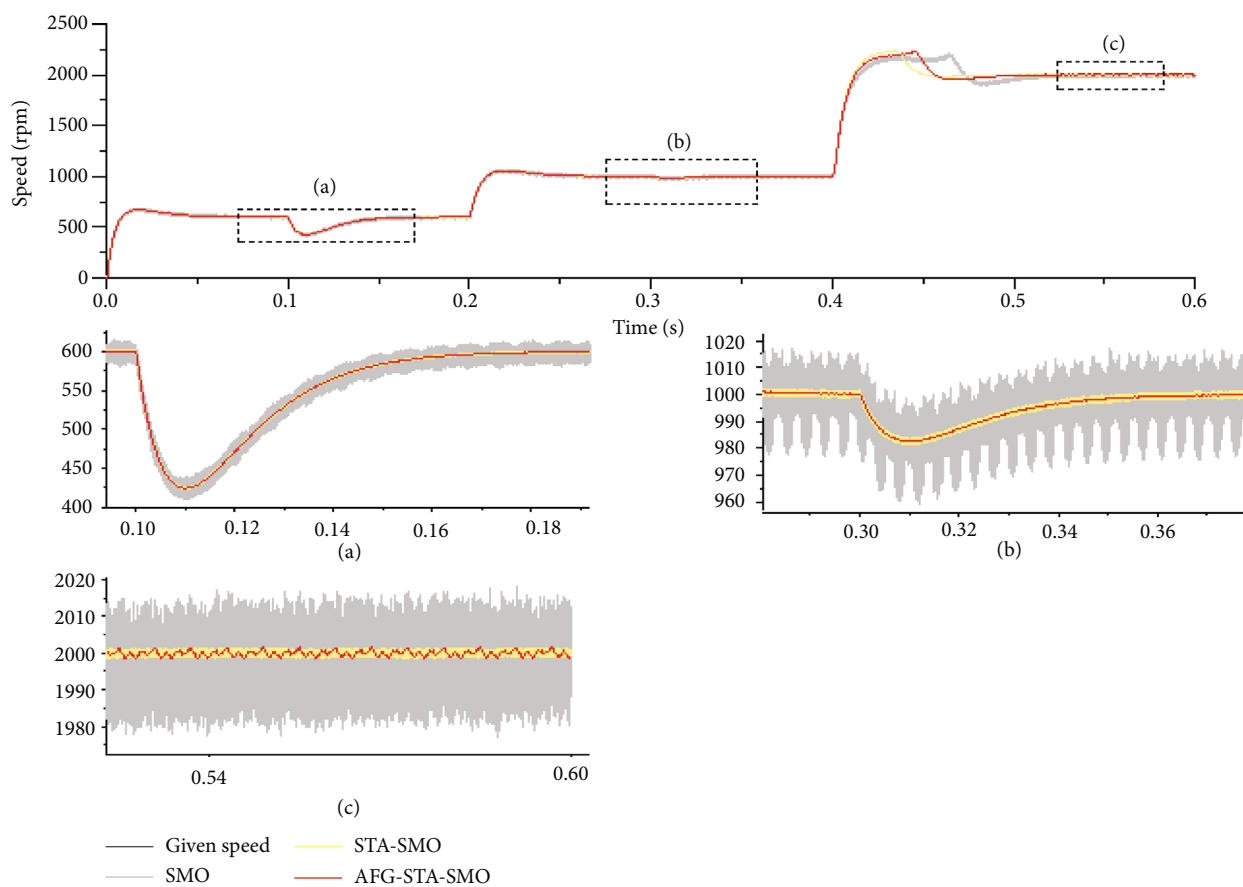Figure 10: Rotor position error diagram.
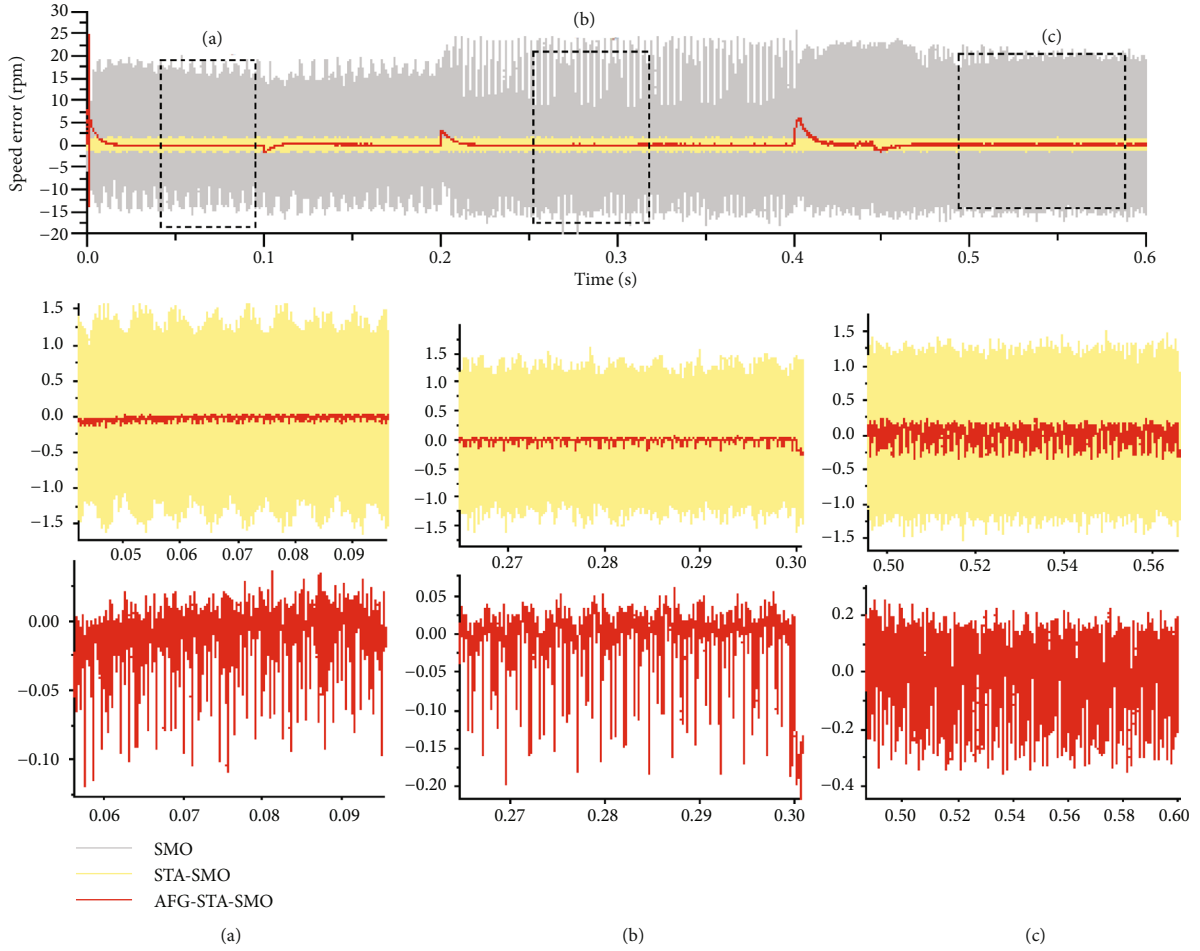

Figure 11: Speed response diagram.

FIGURE 12: Speed error response diagram.

fixed feedback gain are $k_1 = 5000$, $k_2 = 700$, and $l = 500$. The control system parameters of PMSM based on AFG-STA-SMO are as follows: $k_1 = 5000$, $k_2 = 700$, and $\delta = 0.5$.

5.1. Improved AFG-STA-SMO Simulation.
When the SPMSM is under sensor-less control, the rotor position and speed signals are hidden in the back-EMF observation value. This paper mainly uses a super spiral sliding mode observer to extract rotor position and speed information from the back-EMF observations. Although this strategy can avoid introducing low-pass filters, it cannot effectively filter the back-EMF ripple component. In order to obtain a smooth back-EMF signal, this article introduces a Kalman filter to filter the back-EMF signal again. The simulation result is shown in Figure 5. To verify the effectiveness of the strategy proposed in this paper, the motor adopts a no-load starting mode with a given speed of 100 r/min. At 0.2 s, the speed is increased to 300 r/min; at 0.4 s, the speed is increased to 1100 r/min. Figure 5 shows the back-EMF response waveforms under two different strategies.

Table 2 shows that the back-EMF fluctuations of the improved strategy proposed in this paper at three different speed stages are 50%, 8%, and 2.5% of the back-EMF fluctuations of the unimproved strategy, respectively.(Ea1: AFG-STA-SMO without Kalman filter; Ea2: AFG-STA-SMO with

TABLE 4: Speed Error.

Result	SMO	STA-SMO	AFG-STA-SMO	Unit
Error (a)	35	3	0.15	Rpm
Error (b)	45	3	0.25	Rpm
Error (c)	40	3	0.6	Rpm

Kalman filter). By analyzing Figure 5 and the above results, it can be seen that the introduction of the Kalman filter can effectively suppress the ripple component in the back-EMF observation value, thereby obtaining a smoother back-EMF observation signal and further improving the accuracy of the rotor position estimation.

5.2. Verification of Adaptive Feedback Gain Adjustment.
In order to verify the validity of the adaptive law proposed in this article, the motor adopts a no-load starting mode with a given speed of 100 r/min. At 0.2 s, the speed is increased to 300 r/min; at 0.4 s, the speed is increased to 1100 r/min.

From Figures 6 and 7 and Table 3, it is shown that the speed errors of AFG-STA-SMO at three different speed stages are 54.26%, 35.45%, and 100% of STA-SMO speed errors with fixed feedback gain (FFG-STA-SMO), respectively.

TABLE 5: THD and back-EMF fluctuation.

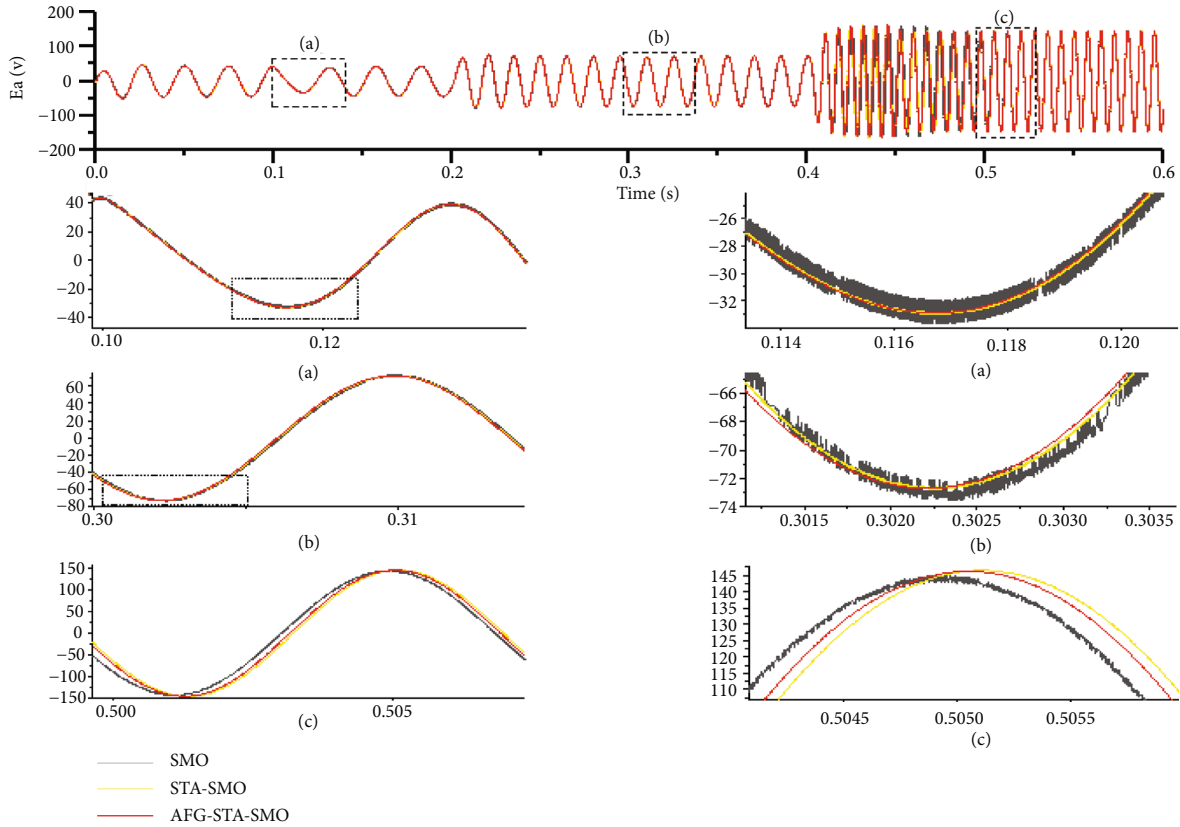| Result	SMO	STA-SMO	AFG-STA-SMO	Unit
THD (a)	8.92	8.62	8.68	%
THD (b)	1.84	1.27	1.24	%
THD (c)	1.50	4.28	3.84	%
Back-EMF fluctuation (a)	-31.8~-33.725	-32.92~-33	-32.91~-32.92	V
Back-EMF fluctuation (b)	-72~-72.36	-72.6~-72.76	-72.685~-72.695	V
Back-EMF fluctuation (c)	143~145	146.35~146.6	46.22~146.23	V

FIGURE 13: Response diagram of back-EMF.

Figures 8–10 and Table 3 show that the rotor position errors of AFG-STA-SMO at three different speed stages are 16.28%, 24.32%, and 100% of FFG-STA-SMO rotor position errors, respectively.

From the above results and the simulation diagram analysis, it can be seen that under the action of the fixed feedback gain, when the motor is running at 1000 r/min, the speed error and rotor position error are minimal. As the speed continues to decrease, the speed error and rotor position error are gradually increasing. After introducing adaptive control, the feedback gain is proportional to the speed. During the entire speed change process, the speed errors are less than the speed errors of FFG-STA-SMO; the estimated rotor position and the actual rotor position are the same, enhancing the system stability.

5.3. Medium and High-Speed Verification Analysis. In order to verify the advantages of the control strategy proposed in

this paper in medium and high-speed operation, the motor adopts a no-load starting mode, and the given speed is 600 r/min. At 0.1 s, load torque of 4.5 N·m is applied; at 0.2 s, the speed rises to 1000 r/min; at the first variable speed stable operation stage (0.3 s), the load torque changes to 5 N·m; at 0.4 s, the speed is increased to 2000 r/min.

From Figure 11, the application of 4.5 N·m and 5 N·m of load torque in 0.1 s and 0.3 s, respectively, causes a slight speed drop in the actual speed. At this time, the estimated speed is still tracking the actual speed, and there is also a slight speed drop that has proved the feasibility and effectiveness of these three control strategies.

Figure 12 and Table 4 show that the speed errors of AFG-STA-SMO at three different speed stages are 0.429%, 0.556%, and 1.5% of SMO speed errors, respectively. The speed errors of AFG-STA-SMO at three different speed stages are 5%, 8.3%, and 20% of STA-SMO speed errors, respectively.
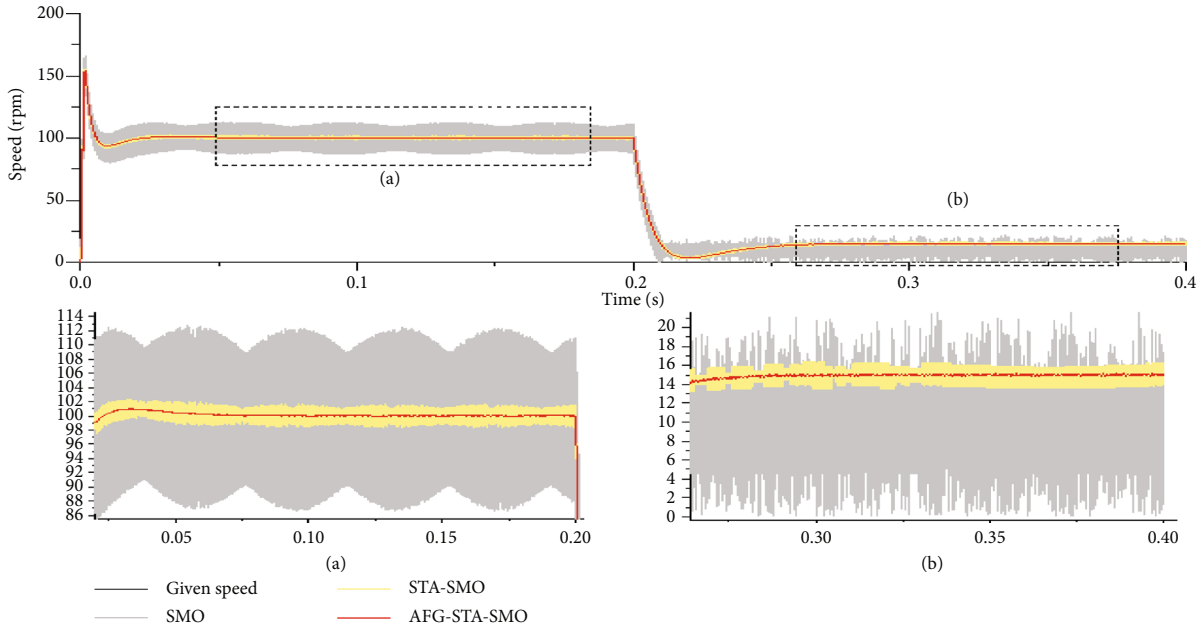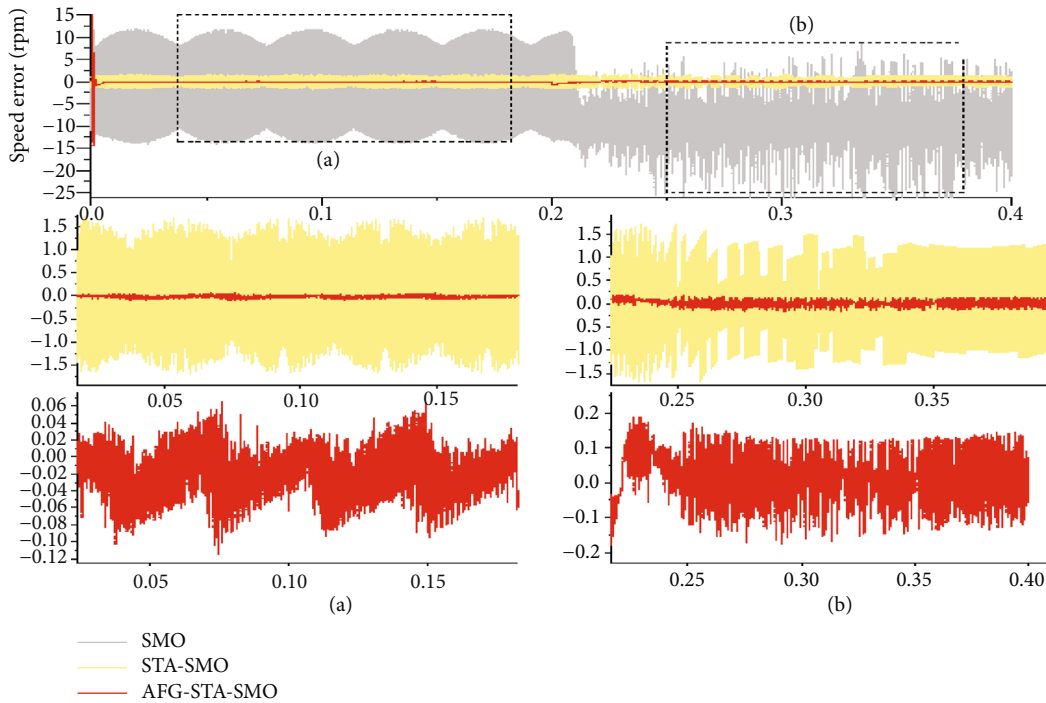
Figure 14: Response diagram of speed.

Figure 15: Response diagram of speed error.

From the above results and simulation waveform analysis, it can be seen that the speed error of SMO increases with the continuous increase of the speed; the sliding mode chattering is serious, which is not conducive to the system's stable operation of the system. There is an obvious lag phenomenon in the speed rise of 2000 r/min. Compared with SMO, STA-SMO effectively suppresses sliding mode chattering due to the characteristics of high-order sliding

Table 6: Speed error and rotor position error.

Result	SMO	STA-SMO	AFG-STA-SMO	Unit
Error (a)	26	3.2	0.17	Rpm
Error (b)	35	3.2	0.4	Rpm
Position error (a)	0.3	0.03	0.0016	Deg
Position error (b)	0.8	0.2	0.006	Deg

Real rotor position
Estimated rotor position

FIGURE 16: The rotor position change of traditional SMO.
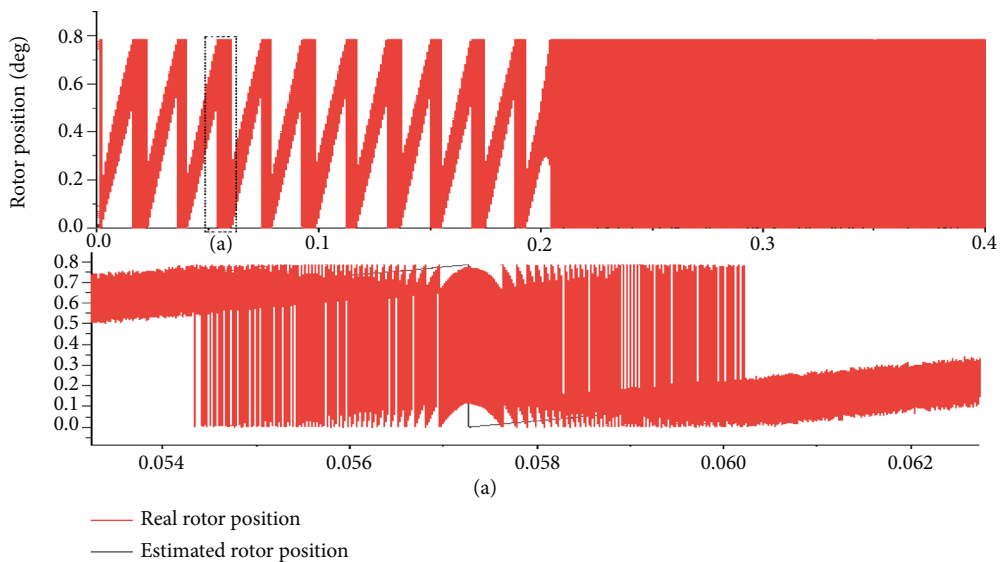

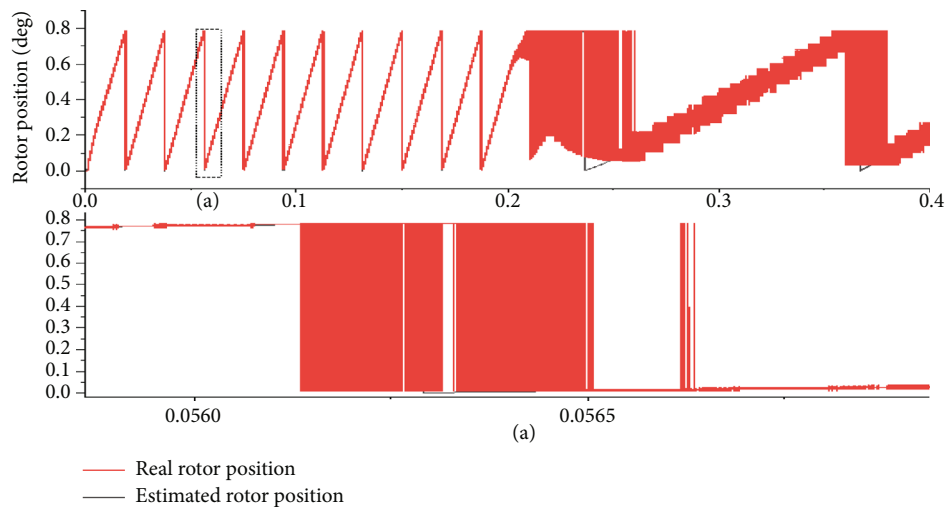Real rotor position
Estimated rotor position

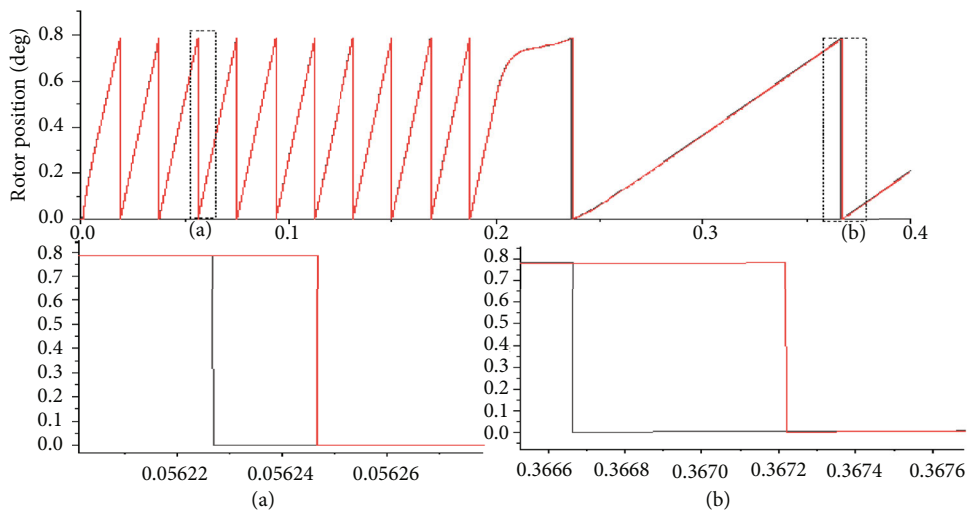FIGURE 17: The rotor position change of traditional STA-SMO.

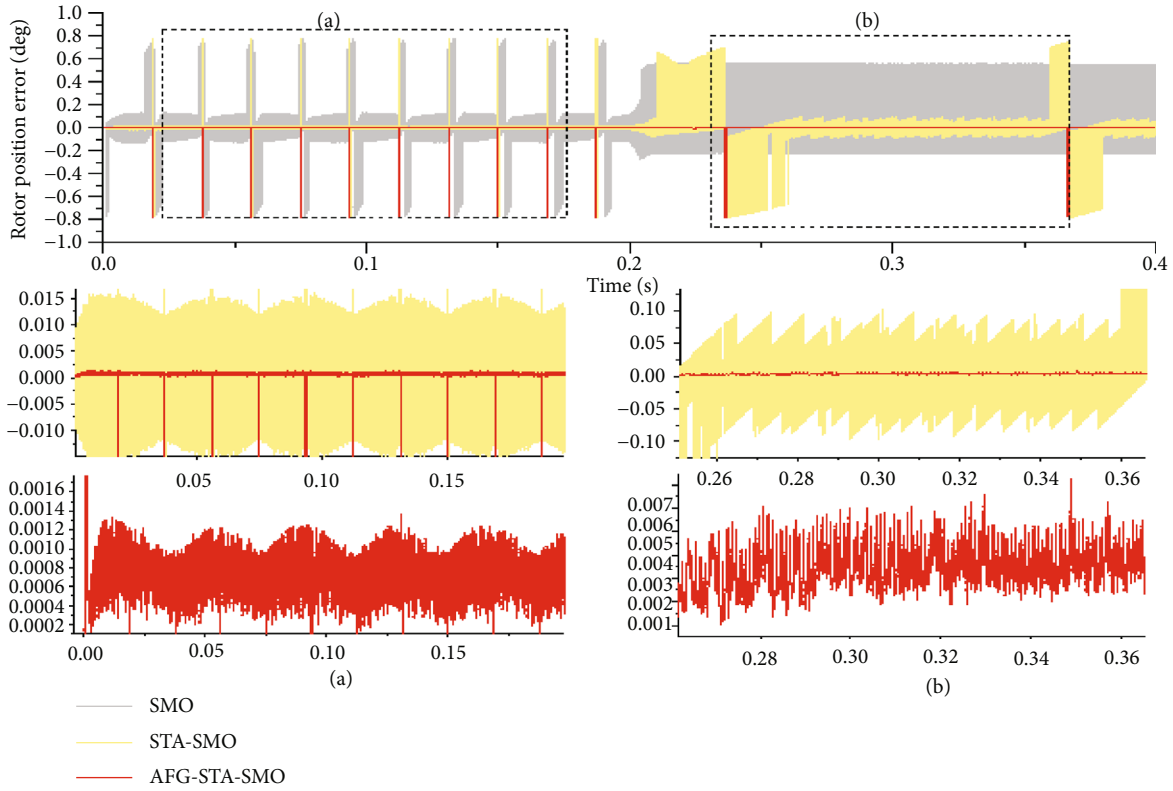FIGURE 18: The rotor position change of AFG-STA-SMO.

Figure 19: Rotor position error of the response diagram.

mode and integral operation function and reduces the speed error, thereby avoiding the lag phenomenon caused by introducing low-pass filters. Compared with STA-SMO, the proposed control strategy is based on STA-SMO, and the Kalman filter and adaptive control are introduced to suppress further sliding mode chattering so that the speed error is minimized, and the change of speed error is small when coping with external disturbance.

Table 5 shows that the back-EMF fluctuations of AFG-STA-SMO at three different speed stages are 0.75%, 2.78%, and 0.5% of the back-EMF fluctuations of SMO, respectively. The back-EMF fluctuations of AFG-STA-SMO at three different speed stages are 12.5%, 6.25%, and 4% of the back-EMF fluctuations of STA-SMO, respectively.

Figure 13, Table 5, and the above analysis of the results show that the SMO's back-EMF has a lag due to the low-pass filter's introduction, which affects the system's response speed. However, STA-SMO avoids this phenomenon due to integral operation. Although these two control strategies can effectively filter out the harmonic components, there are many ripple components in the back-EMF value. If they are directly used to estimate the rotor position, they will inevitably cause larger errors. Therefore, the Kalman filter is introduced to perform filtering processing to effectively suppress the back-EMF ripple component, thereby obtaining a smoother back-EMF signal and improving control accuracy.

5.4. Low-Speed Verification Analysis. In order to verify the advantages of the sensor-less algorithm-controlled speed control system in low-speed operation situations, simulation

experiments are carried out in low-speed areas. The motor adopts a no-load start mode, and the given speed is 100 r/min. At 0.2 s, the speed suddenly changes to 15 r/min.

Figure 14, Figure 15, and Table 6 show that the speed errors of AFG-STA-SMO at two different speed stages are 0.6534% and 1.1423% of the SMO speed errors, respectively. The speed errors of AFG-STA-SMO at two different speed stages are 5.3125% and 12.5% of STA-SMO speed errors, respectively.

Figures 16–19 and Table 6 show that the rotor position errors of AFG-STA-SMO at two different speed stages are 0.53% and 0.75% of SMO rotor position errors, respectively. The rotor position errors of AFG-STA-SMO at two different speed stages are 5.33% and 3% of STA-SMO rotor position errors, respectively.

From the above results and simulation diagram analysis, it can be seen that when the motor runs at low speed, the chattering of SMO speed waveform and rotor estimated position waveform is serious, and the speed error and rotor position error are the largest. When the speed is reduced to 15 r/min, the system almost cannot operate normally. Compared with SMO, STA-SMO suppresses sliding mode chattering to a certain extent due to the characteristics of high-order sliding mode and the function of the integrator. However, the back-EMF is too small when the motor runs at low speed, resulting in rotor position error and speed error deviation. It is not conducive to the stable operation of the system. Compared with STA-SMO, the estimated rotor position of AFG-STA-SMO is basically the same as the actual rotor position due to the introduction of

equivalent feedback gain, and the speed error and rotor position error are minimal, thus avoiding the situation that the rotor position error is too large due to the small back-EMF.

6. Conclusion

This paper presents a sensor-less control strategy for STA-SMO with adaptive feedback gain. The adaptive law, which compensates for rotor angle estimation error by adjusting the feedback gain coefficient online, effectively solves the motor's problems of low-speed operation difficulty and serious sliding mode chattering.

Secondly, the Kalman filter is used to reduce the ripple component in the back-EMF. The smoother back-EMF signal is obtained, and the control precision of the algorithm is further improved.

The whole system realizes the full speed domain operation; the dynamic response is fast and has the strong disturbance resistance when dealing with the external disturbance. In the next step, the author intends to implement the proposed control strategy with high-performance DSP chip in order to improve the operation time extension caused by the increasing complexity of the control system.

Data Availability

The data supporting the findings of this study are available within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] L. Zhang, S. Wang, and J. Bai, "Fast-super-twisting sliding mode speed loop control of permanent magnet synchronous motor based on SVM-DTC," *IEICE Electronics Express*, vol. 18, no. 1, article 20200375, 2021.

[2] B. K. Bose, "Power electronics and motion control-technology status and recent trends," *IEEE Transactions on Industry Applications*, vol. 29, no. 5, pp. 902–909, 1993.

[3] F. Dongxue, X. Zhao, and H. Yuan, "High-precision motion control method for permanent magnet linear synchronous motor," *IEICE Electronics Express*, vol. 18, article 20210097, 2021.

[4] X. Zhao and D. Fu, "Adaptive neural network nonsingular fast terminal sliding mode control for permanent magnet linear synchronous motor," *IEEE Access*, vol. 7, pp. 180361–180372, 2019.

[5] T. M. Jahns and V. Blasko, "Recent advances in power electronics technology for industrial and traction machine drives," *Proceedings of the IEEE*, vol. 89, no. 6, pp. 963–975, 2001.

[6] G. Foo and M. F. Rahman, "Sensorless sliding-mode MTPA control of an IPM synchronous motor drive using a sliding-mode observer and HF signal injection," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1270–1278, 2010.

[7] L. Ding, Y. W. Li, and N. R. Zargari, "Discrete-time SMO sensorless control of current source converter-fed PMSM drives with low switching frequency," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2120–2129, 2021.

[8] Q. Zhu, Z. Li, and X. Tan, "Sensors fault diagnosis and active fault-tolerant control for PMSM drive systems based on a composite sliding mode observer," *Energies*, vol. 12, no. 9, p. 1695, 2019.

[9] W. Zhu, "A position sensorless control strategy for SPMSM based on an improved sliding mode observer," *IOP Conference Series Earth and Environmental Science*, vol. 252, article 032135, 2019.

[10] K. Prabhakaran, K. Anbalagan, and F. Blaabjerg, "Laboratory implementation of electromagnetic torque based MRAS speed estimator for sensorless SMPMSM drive," *Electronics Letters*, vol. 55, no. 21, pp. 1145–1147, 2019.

[11] E. Dehghan-Azad, S. Gadoue, D. Atkinson, H. Slater, P. Barrass, and F. Blaabjerg, "Sensorless control of IM based on stator-voltage MRAS for limp-home EV applications," *IEEE Transactions on Power Electronics*, vol. 33, no. 3, pp. 1911–1921, 2018.

[12] Z. Yin, G. Li, Y. Zhang, and J. Liu, "Symmetric-strong-tracking-extended-Kalman-filter-based sensorless control of induction motor drives for modeling error reduction," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 650–662, 2019.

[13] J. Sun, G. Cao, S. Huang, Y. Peng, J. He, and Q. Qian, "Sliding-mode-observer-based position estimation for sensorless control of the planar switched reluctance motor," *IEEE Access*, vol. 7, pp. 61034–61045, 2019.

[14] C. M. Verrelli, S. Bifaretti, E. Carfagna et al., "Speed sensor fault tolerant PMSM machines: from position-sensorless to sensorless control," *IEEE Transactions on Industry Applications*, vol. 55, no. 4, pp. 3946–3954, 2019.

[15] K. Paponpen and M. Konghirun, "An improved sliding mode observer for speed sensorless vector control drive of PMSM," in *2006 CES/IEEE 5th international power electronics and motion control conference*, pp. 1–5, Shanghai, 2006.

[16] L. Guo, X. Zhang, S. Yang, Z. Xie, L. Qi, and L. Wang, "Super-twisting sliding mode observer based speed sensorless torque control for PMSG used in wind turbines," in *2015 9th International Conference on Power Electronics and ECCE Asia (ICPE-ECCE Asia)*, pp. 2457–2462, Seoul, 2015.

[17] A. Levant, "Principles of 2-sliding mode design," *Automatica*, vol. 43, no. 4, pp. 576–586, 2007.

[18] D. Wang, J. Liu, S. Miao et al., "Rotor position estimation method for permanent magnet synchronous motor based on super-twisting sliding mode observer," in *2018 37th Chinese Control Conference (CCC)*, pp. 5634–5638, Wuhan, 2018.

[19] S. Chi, Z. Zhang, and L. Xu, "Sliding-mode sensorless control of direct-drive PM synchronous motors for washing machine applications," *IEEE Transactions on Industry Applications*, vol. 45, no. 2, pp. 582–590, 2009.

[20] S. Chi, L. Xu, and Z. Zhang, "Sliding mode sensorless control of PM synchronous motor for direct-driven washing machines," in *Conference Record of the 2006 IEEE industry applications conference forty-first IAS annual meeting*, pp. 873–879, Tampa, FL, 2006.

[21] S. Chi, Z. Zhang, and L. Xu, "A novel sliding mode observer with adaptive feedback gain for PMSM sensorless vector control," in *2007 IEEE power electronics specialists conference*, pp. 2579–2585, Orlando, FL, 2007.

[22] F. Zhao, L. U. O. Wen, G. A. O. Fengyang, and Y. U. Jiale, "Based on the fuzzy sliding mode control and two stage filtering all-digital fuzzy sliding mode observer is improved control strategy (English)," *Journal of Measurement Science and Instrumentation*, vol. 1, no. 11, 2020.

[23] Y. Fan, L. Zhang, M. Cheng, and K. T. Chau, "Sensorless SVPWM-FADTC of a new flux-modulated permanent-magnet wheel motor based on a wide-speed sliding mode observer," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 5, pp. 3143–3151, 2015.

[24] C. Junjie, *Sensorless Control of Permanent Magnet Synchronous Motor Based on Improved Sliding Mode Observer*, Shanghai Institute of Electrical Engineering, 2019.

[25] G. Wang, Z. Li, G. Zhang, Y. Yu, and D. Xu, "Quadrature PLL-based high-order sliding-mode observer for IPMSM sensorless control with online MTPA control strategy," *IEEE Transactions on Energy Conversion*, vol. 28, no. 1, pp. 214–224, 2013.